
Intermediate-energy Coulomb excitation with the PreSPEC-AGATA setup

Coulomb-Anregung bei mittleren Energien am PreSPEC-AGATA Aufbau

Vom Fachbereich Physik der Technischen Universität Darmstadt

zur Erlangung des Grades eines Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigte Dissertation von Michael Reese aus Hildesheim

Darmstadt 2018 — D 17



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Physik
Institut für Kernphysik
AG Pietralla

This work was supported by



Bundesministerium
für Bildung
und Forschung

FKZ: 06DA90401,05P12RDFN8

and

HIC | **FAIR**
for
Helmholtz International Center

Intermediate-energy Coulomb excitation with the PreSPEC-AGATA setup
Coulomb-Anregung bei mittleren Energien am PreSPEC-AGATA Aufbau

Genehmigte Dissertation von Michael Reese aus Hildesheim

1. Gutachten: Prof. Dr. Dr. h.c. Norbert Pietralla
2. Gutachten: Prof. Dr. Thomas Aumann

Tag der Einreichung: 18.4.2017

Tag der Prüfung: 14.6.2017

Dokumentenversionsnummer: 1.1.0

Darmstadt — D 17

Bitte zitieren Sie dieses Dokument als:

URN: urn:nbn:de:tuda-tuprints-73279

URL: <http://tuprints.ulb.tu-darmstadt.de/7327>

Dieses Dokument wird bereitgestellt von tuprints,

E-Publishing-Service der TU Darmstadt

<http://tuprints.ulb.tu-darmstadt.de>

tuprints@ulb.tu-darmstadt.de



Die Veröffentlichung steht unter folgender Creative Commons Lizenz:

Namensnennung – nicht kommerziell – Weitergabe unter gleichen Bedingungen 4.0 International

<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.de>

Lebenslauf (curriculum vitae)

Persönliche Daten

Name: Michael
Vorname: Reese
Geburtsdatum: 27.04.1983
Geburtsort: Hildesheim

Berufserfahrung

ab 09.2016 VHDL / C++ Entwickler
GSI Helmholtzzentrum für Schwerionenforschung GmbH, Darmstadt
01.2016 - 08.2015 Hardware/Firmware/Software Entwickler
Gesellschaft für Forschung und Entwicklungsservice mbH (GFE)
08.2015 - 12.2015 Wissenschaftlicher Mitarbeiter
GSI Helmholtzzentrum für Schwerionenforschung GmbH, Darmstadt
12.2014 - 12.2015 Wissenschaftlicher Berater in Teilzeit
Gesellschaft für Forschung und Entwicklungsservice mbH (GFE)
08.2009 - 06.2015 Wissenschaftlicher Mitarbeiter
Technischen Universität Darmstadt

Schul- und Berufsbildung

04.2003 - 06.2009 Diplomstudiengang Physik
Technischen Universität Darmstadt
09.2006 - 06.2007 SOCRATES/Erasmus - Austauschprogramm
Institut National Polytechnique de Grenoble, Frankreich
07.2002 - 03.2003 Grundwehrdienst
2./Panzergrenadierlehrbataillon 92., Munster
1995 - 2002 Bischöfliches Gymnasium Josephinum in Hildesheim
1993 - 1995 St.-Augustinusschule Hildesheim
1989 - 1993 St.-Mauritiusschule Hildesheim

*In science you want to say something that nobody knew before,
in words which everyone can understand.*

Paul Adrien Maurice Dirac

Acknowledgments

It is done! Finally, I managed to write a document that meets my personal requirements. It took me eight years, and that is just too much time to feel only positive about it. There have been good times, but there have also been stressful and desperate times. More than once I was close to giving up, and I was capable of finalizing this work only after leaving the scientific world, retrieving my motivation by working on projects where I saw my talents being more promoted, and my way of working being more appreciated. During all these times I received financial and motivational support that helped me to finish this work. Here, I would like to express my gratitude and appreciation to these people and organizations:

Prof. Dr. Dr. h.c. **Norbert Pietralla** for allowing me to stay in his group for such a long time, and who also let me go when I felt it was the right thing to do. He is one of the too few persons who managed to get me excited about nuclear structure, and who seems to be doing science out of honest curiosity. Dr. **Jürgen Gerl** for letting me work at GSI, and for giving me the unique opportunity to do my first steps into industry research and prototype development. Dr. **Tobias Engert** for hiring me at GFE, giving me the opportunity to finish this thesis while working on exciting hardware development projects. Dr. **Liliana Cortés** for being such a good friend and a continuous source of motivation, and for all the detector work. Dr. **Plamen Boutachkov** for being a very competent, friendly, and open minded scientist. Dr. **Stefan Typel** for helping me to understand the multipole expansion to a point where I could program it. Dr. **Oliver Möller** who turned my interest towards nuclear structure many years ago. **Christian Röder** and Dr. **Bastian Löher** for their enthusiasm in programming, hardware, and technology. I enjoyed working with Dr. **Sudipta Saha**, Dr. **Guangshun Li**, Dr. **Tobias Habermann**, **Marc Lettmann**, Dr. **Jacob Beller**, Dr. **Andreas Krugmann**, Dr. **Jelena Vesić**, **Tuğba Arici**, **Henning Schaffner**, Dr. **Giulia Guastalla**, Dr. **César Domingo Pardo**, Dr. **Nataša Lalović**, **Pavlos Koseoglou**, Dr. **Christoph Kremer**, **Waldemar Witt**, Dr. **Cesar Lizarazo**. I thank **Silke Noll** for significantly reducing the number of typos and spelling errors in the manuscript.

The AGATA, PreSPEC, LYCCA collaborations for providing the instruments and allowing me work with the acquired data.

All authors of free software and providers of free knowledge. The whole work was accomplished using mainly free software tools, written and provided by a large number of contributors.

I acknowledge the financial support from the Bundesministerium für Bildung und Forschung (BMBF), The Helmholtz International Center for FAIR (HIC for FAIR), and I would like to thank the Helmholtz Graduate School for Hadron and Ion Research (HGS-HIRE) for the lecture weeks, the power weeks, and the HGS-HIRE excellence award 2013.

Abstract

This thesis is about the improvement of software tools and methods for analyzing high resolution γ -ray spectroscopy data of intermediate energy Coulomb excitation experiments at the PreSPEC-AGATA setup. In the past, pioneering experiments have been done in the RISING campaign at GSI. The PreSPEC-AGATA setup has been the next step in the evolution of this type of experiments, with better and more complex instrumentation, leading to higher sensitivity and resolution. This work is relevant for the preparation of the even more demanding future HISPEC setup at the upcoming FAIR facility. HISPEC experiments will study low-lying excited states of nuclei far from stability, in order to test the validity of nuclear structure models in that area of the nuclear chart.

More complex instrumentation results in more complex data analysis programs. In order to deal with the complexity involved in developing such software for the upcoming experiments, a more systematic approach to the development process is needed. Within this work, a software framework was developed to facilitate a systematic and clean software design. This might be helpful for upcoming NUSTAR experiments at FAIR, where a large number of independent detector systems have to be integrated into a big setup. The design of the framework is discussed in the second chapter of this thesis.

One important reaction mechanism for the HISPEC experiments is Coulomb excitation at energies of around 150 MeV/u. Because of the purely electromagnetic nature of the interaction, Coulomb excitation is described well by theoretical models. In the third chapter of this thesis, a numerical simulation program for relativistic ion-ion collisions was developed that solves the coupled channel equations for electric Coulomb excitation.

In the fourth chapter, the relevant detector systems of the PreSPEC setup are presented, and the calibration of them is discussed. A main aspect of the work was the development of improved ways to analyze data from these devices, and, if possible to provide semi-automatic implementations for their calibration.

In the last chapter, the total efficiency of the PreSPEC-setup was determined for a typical Coulomb excitation experiment, and the possibility to measure projectile excitation cross-sections relative to a target excitation cross-section is demonstrated.

Zusammenfassung

Der Fokus dieser Arbeit liegt in der Verbesserung von Programmen und Methoden zur Analyse von hochauflösenden γ -Spektroskopie-Daten der Coulomb-Anregungsexperimente, die mit dem PreSPEC-AGATA-Aufbau bei mittleren Energien durchgeführt wurden. Erste derartige Experimente wurden während der RISING Kampagne an der GSI durchgeführt. Der PreSPEC-AGATA Experimentaufbau stellte die nächste Stufe in der Entwicklung dieser Experimente dar, bei dem durch Verbesserungen in der Instrumentierung eine Erhöhung der Sensitivität und der Auflösung ermöglicht wird. Der Inhalt dieser Arbeit ist ein Beitrag zur Vorbereitung von weiter verbesserten Experimenten mit exotischen Atomkernen an dem geplanten HISPEC-Aufbau an der im Bau befindlichen FAIR Beschleunigeranlage. In diesen Experimenten sollen niedrig liegende, angeregte Zustände in Atomkernen jenseits des Tals der Stabilität untersucht werden.

Verbesserte Instrumente führen zu größerer Komplexität bei der Durchführung und Datenanalyse der Experimente. Eine systematische Vorgehensweise bei der Softwareentwicklung ist nötig, um die damit verbundene Komplexität zu bewältigen. Im Rahmen dieser Arbeit wurde ein Software-Framework entwickelt, dass bei der sauberen und systematischen Erstellung von Datenanalyse-Software hilft. Solch ein Framework kann insbesondere bei kommenden NUSTAR Experimenten hilfreich sein, bei denen viele unabhängig voneinander entwickelte Detektorsysteme in einen großen Aufbau integriert werden sollen. Das Design des Software-Frameworks wird im zweiten Kapitel dieser Arbeit vorgestellt.

Ein wichtiger Reaktionsmechanismus bei den HISPEC Experimenten ist die Coulomb-Anregung bei kinetischen Energien von etwa 150 MeV/u. Coulomb-Anregung kann als rein elektromagnetischer Prozess gut theoretisch beschrieben werden. Im dritten Kapitel der Arbeit wurde ein numerisches Simulationsprogramm für relativistische Schwerionenkollisionen entwickelt, welches die gekoppelten Differenzialgleichungen für die elektrische Coulomb-Anregung löst.

Im vierten Kapitel wird der PreSPEC-AGATA Aufbau mit seinen Teilsystemen präsentiert und die Kalibrierung der Einzelsysteme wird diskutiert. Ein wichtiger Aspekt war dabei, wenn möglich, semi-automatische Algorithmen zu implementieren, die die Kalibrierung vereinfachen.

Im letzten Kapitel wird die Gesamteffizienz des PreSPEC-Aufbaus für ein typisches Coulombanregungs-Experiment bestimmt sowie die Möglichkeit der relativen Messung von Target- und Projektilanregungsquerschnitten demonstriert.

Contents

1. Introduction	8
2. Software framework for event-based data analysis	11
2.1. Generic tasks and customization points in event-based data analysis	11
2.2. Design of the software framework	15
2.2.1. Data flow description with the Elder-DSL	15
2.2.2. Adding analysis algorithms	22
2.2.3. Adding module unpackers	26
2.2.4. The Elder API for analysis programs	28
2.3. Conclusions	31
3. Nuclear collision dynamics and Coulomb excitation	33
3.1. Relativistic dynamics	33
3.1.1. Equations of motion	34
3.1.2. Retarded electromagnetic fields	34
3.2. Numerical scattering solutions of the relativistic Kepler problem	36
3.2.1. Implementation	36
3.2.2. Verification of the simulation	37
3.3. Coulomb excitation	43
3.3.1. Time dependent perturbation theory	43
3.3.2. Numerical solution	45
3.3.3. Analytic solution for the straight line approximation	47
3.3.4. Multi-step Coulomb excitation	48
3.3.5. Cross sections	49
3.3.6. DWEIKO	50
3.3.7. Summary	50
4. The PreSPEC experimental setup	52
4.1. The Lund-York-Cologne Calorimeter Array (LYCCA)	55
4.1.1. Overview of the detector system	55
4.1.2. Principle of operation	57
4.1.3. Time-of-flight detectors	60
4.1.4. DSSSD calibration	67
4.1.5. Absolute calibration of the LYCCA DSSSD detectors	72
4.1.6. CsI calibration	73
4.1.7. Bayesian LYCCA isotope identification	74
4.2. The Advanced Gamma Tracking Array (AGATA)	81
4.2.1. Calibration	82
4.2.2. Simplified γ tracking scheme	83

4.3. PreSPEC data acquisition	83
4.3.1. MBS	83
4.3.2. AGATA	84
4.3.3. Coupling of AGATA to the VME	86
5. Analysis of the S424 experiment	88
5.1. Experiment parameters	88
5.2. Qualitative description of processes in the target area	89
5.3. Properties of γ -ray spectra	92
5.4. Doppler Effect	93
5.5. Scattering angle and angular straggling	96
5.6. Coulomb excitation of ^{80}Kr	96
5.6.1. Background reduction	96
5.6.2. Doppler corrected γ spectra	98
5.6.3. Angular distribution of γ -rays	98
5.6.4. Comparison with theory	101
5.6.5. Differential cross-section	101
5.6.6. Measured total PreSPEC-AGATA efficiency	103
5.6.7. Efficiency considerations	104
5.6.8. Nuclear excitation from particle detectors	109
5.6.9. Discussion	110
A. Getting started with PreSPEC data analysis	112
A.1. Raw data access	112
A.2. Match MBS and AGATA data files	113
A.3. Access to the software tools	113
A.4. Elder-DSL script for the experiment	114
A.5. Online PreSPEC forum	114
A.6. Root trees	114
A.7. Developments	114
A.8. AGATA PSA replay	115

1 Introduction

The atomic nucleus is a self-bound many-body system of strongly interacting N neutrons and Z protons. It has a positive charge of $q = eZ$, a mass $m \approx Au$ which is proportional to the number of nucleons $A = Z + N$, and a radius $R \approx 1.2 \text{ fm} \cdot A^{1/3}$ [1, 2, 3]. In our surrounding environment most nuclei are in the ground state, but they have a rich amount of excited states including single or few nucleon excitations as well as collective motions involving many or all nucleons. Each state in a nucleus can be characterized by an excitation energy E , spin and parity quantum numbers J^π , and electric and magnetic multipole moments describing electrical charge and current distributions and spin orientations of nucleons. Any non-stationary charge or current distribution interacts with electromagnetic fields, and the nucleus is no exception. External electromagnetic fields of a certain frequency can cause transitions between different states.

Information about internal charge and current distributions can be obtained by studying the response of nuclei to electromagnetic fields. If it is possible to expose a nucleus to a well-known time dependent electromagnetic field, and if it is further possible to measure the probability of different nuclear excitation modes that are populated after the application of the field, the charge and current distributions of these modes can be deduced by employing the theory of electromagnetism. The calculation can be done with a high accuracy and relatively small computational effort by means of perturbation theory due to the small electromagnetic coupling constant $\alpha \approx 1/137$.

An appropriate time-dependent electromagnetic field for the excitation of a nucleus is created during the fly-by of a charged particle in a nuclear collision process. If the distance of the two colliding particles is larger than the range of nuclear forces throughout the entire collision trajectory, the process is undisturbed by nuclear interactions that are far more difficult to calculate than electromagnetic interactions. As a result, each of the two colliding nuclei can be in an excited state induced by a very well known time dependent electromagnetic field. This technique of producing nuclei in excited states is called Coulomb excitation [4, 5].

One way of observing the final state of the nucleus after the excitation process is the measurement of the γ -radiation, which is emitted from a nucleus while it decays back into the ground state [6]. Spectroscopy of the γ -radiation allows to deduce the excitation energy of the nucleus and, in many cases, the exact excited state that was populated after the excitation process. Coulomb excitation in conjunction with γ -ray spectroscopy is applied since the 1950s [7], and a large body of data has been collected since then using beams and targets of naturally occurring stable isotopes.

In the late 1970s and 1980s, experiments with isotopes further away from the line of stability were done. A set of anomalies were observed in the semi-magic exotic neutron-rich nuclei with $Z \ll N = 20$, including an unexpected ground state spin and deformation of ^{31}Na [8] and the extremely low excitation energy of the 2_1^+ state of ^{32}Mg [9]. This observation was in disagreement [10] with predictions of the popular nuclear shell model [11, 12], an effective theory that accounts for the occurrence of exceptionally tightly bound isotopes having particular numbers of protons and neutrons, the so-called magic numbers [13]. In the context of this

model, the phenomenon of changed magic numbers in certain exotic isotopes is nowadays called shell evolution [14]. This was one important results leading to an increased effort to support experiments with even more exotic isotopes.

In the 1970s, the in-flight technique for radioactive ion beam (RIB) production was developed at the Bevalac accelerator at the Lawrence Berkeley Laboratory (LBL, now LBNL) [15]. In-flight RIB facilities use reactions between stable isotopes to produce a cocktail beam of fast moving exotic nuclei, of which the isotope of interest is selected in a magnetic separator and guided to the experiment.

In the 1990s, experiments at National Superconducting Cyclotron Laboratory (NSCL) with the A1200 Separator [16] and at the RIKEN Accelerator Research Facility used radioactive beams from the in-flight method in conjunction with Coulomb excitation and γ -ray spectroscopy with NaI Scintillation detector arrays [17, 18, 19]. Similar experiments were developed around the same time at GSI, using the Fragment Separator (FRS) for in-flight separation and the Crystal Ball NaI-array for γ -ray detection [20]. Because of the high projectile velocity of around $v/c \approx 0.3$ to 0.5 this technique is called *intermediate energy Coulomb excitation*. The high particle velocity leads to many problems such as increased atomic background radiation levels and unwanted fragmentation reactions which make it necessary to identify the outgoing isotope after the reaction. In addition, large Doppler broadening of the measured γ -radiation limits the spectroscopic resolution.

At all the facilities, these experiments were further developed to deal with the mentioned problems. GSI continued in the 2000s with the RISING fast-beam experiments [21], using high-purity Germanium detectors for γ -ray detection in order to improve the resolution of spectroscopic results, and the CATE E - ΔE telescope for identification of the outgoing isotopes.

In the 2010s, the PreSPEC experiments at GSI had a focus on Coulomb excitation and fragmentation reaction studies using in-flight RIBs from the FRS. The next generation γ -ray detection array AGATA [22] was used for γ -spectroscopy and an improved E - ΔE -ToF detector, called LYCCA [23], identified the outgoing isotopes. The PreSPEC experimental campaign was the stage for the work presented in this thesis, as well as a preparation phase for future HISPEC experiments at the FAIR facility that are planned to happen in the 2020s.

The continuous progress in instrumentation was done in collaboration with different institutes in different countries, contributing with various detector systems to the experimental setup. Integrating these detectors into one working system was difficult, especially regarding the data analysis because the interoperability of different data analysis routines was not taken into account during their development. The second chapter of this thesis suggests a technical solution for the integration of data analysis routines for different detector systems into an analysis program for the entire experiment. Such a system could be used in the future to improve the interoperability of the data analysis routines for detector systems, even if the development happens independently by different groups within a large community.

Apart from the integration of the different PreSPEC detectors into one analysis program, most of them were new developments and the procedures for their calibration and analysis were not optimized yet. In the fourth chapter of this work, all relevant detector systems are discussed, with a focus on improvements in the processing of the LYCCA data, including new calibration procedures. The S424 PreSPEC experiment was intended to measure performance parameters of the setup and is discussed in the fifth chapter. Data from the Coulomb excitation part of this experiment were analyzed and the absolute efficiency of the setup is determined. Furthermore,

it is shown that in this kind of experiment the target excitations can be used as normalization reference for the projectile excitation.

An improved numerical calculation for the process of Coulomb excitation at intermediate energies is presented in the third chapter. It allows to determine the accuracy of often used approximations in the analysis of typical PreSPEC Coulomb excitation experiments.

2 Software framework for event-based data analysis

The main experimental data that was analyzed within this thesis was measured with a relatively complicated setup. During the work on this data set, and based on experience with other experimental setups, the necessity for a more systematic treatment of this kind of data became apparent. This chapter contains an independent presentation of a fairly generic software framework called Elder [24] that is the result of a design process in order to simplify the PreSPEC-AGATA data analysis, but also the development of future data analysis software packages. A software infrastructure for these kinds of experiments should provide the following set of features:

1. The system should have a modular design that reflects modular nature of the experiment, including the detectors and the front-end electronics. Truly independent modules require interfaces that support high-level data exchange.
2. It should be simple to extend the functionality of the whole system by adding new independent algorithms and front-end electronics modules as independent building blocks.
3. The composition of these building blocks should be as simple as possible. Especially during the development phase, the changes should also be visible as fast as possible, preferably without compilation of any source code.
4. Visualization of data should be as simple as possible, including the creation of gated histograms with arbitrary conditions.

Existing software systems are inappropriate, because they fail to provide at least one of these essential features. For example, Cracow [25] has a focus on item 4., but customizability and composition of the algorithms is not part of its user interface. Users are confined to one fixed analysis. UCESB [26] has a focus on 1., 2., and 3., but limits itself to the treatment of front-end electronics and the unpacking of the data and does not help with the composition of analysis algorithms for data reduction. NARVAL [27] fulfills items 1. and ostensibly 2. by providing a freely configurable and modular system for data analysis. But it does not define a high-level interface between the modules, and thereby creates an implicit dependency between algorithms (fails item 2.). In addition, the configuration in XML does not fulfill the requirement of simplicity (item 3.), and data visualization is not part of this system at all (item 4.) This chapter describes a modular and user friendly software system was developed and tested that provides all required features.

2.1 Generic tasks and customization points in event-based data analysis

Event-based data analysis is very common in nuclear structure research. It is a process of data reduction, starting with list-mode data that contains events with correlated raw data from different detectors. Typically, the process ends with histograms of highly processed and selected data,

for example, a Doppler corrected γ -ray spectrum that appears in coincidence with a particle of certain charge and mass. It makes sense to spend some time to identify the requirements for analyzing this kind of data, because some aspects of event-based data analysis do not depend on the details of the experimental setup. Within this chapter, these aspects are called “generic tasks”. Each software program for event-based data analysis has to do these tasks in one form or another. It is possible to factorize these generic parts of the analysis process into a reusable software framework. Such a framework will contain solutions to all problems that are common to *any* data analysis software, and will allow for easy customization of all parts that might be specific to a particular analysis or experimental setup. The generic tasks are categorized into four groups: raw data handling, data processing, data flow and data visualization.

Raw data handling

In each recorded event, the actual encoding of the information depends on the acquisition system in use. Consequently, the decoding or unpacking¹ part of the analysis software is not generic and must be a point of customization. However, most of the electronics used in the field of nuclear structure physics is modular. A specific type of ADC module can be found in different experiments, and the encoding of data from that specific module is always the same. Furthermore, the modules are grouped into readout units that correspond to their physical arrangement in crates². These readout units are groups of modules that are handled together by the acquisition system and their encoded raw data is written sequentially into one block of the raw data section of the event. Two generic tasks can be identified that can be implemented in a generic framework.

- Given access to the raw data of a readout unit inside one event, the analysis software has to unpack all data from the modules inside that readout unit in sequence.
- The software has to handle all readout units inside one event systematically. Such a readout unit is often given by the physical arrangements of readout modules in crates

At the same time, it must be possible to customize the specific parts of the data decoding process. There are two points of customization.

- The details of the raw data unpacking depends on the type of modules used and is unknown to the framework. It has to be possible to extend the framework by adding specific module unpacking subprograms, so called *unpackers*.
- The composition of readout units and electronic modules therein is specific to each setup. The software framework has to provide means to compose available unpacking programs into sequential groups.

The first point is so general, that the only way to offer users of the framework full flexibility, is to use a general purpose language for this part of the customization process. As the framework itself is written in C++ the unpackers have to be provided by users in the form of C++ code. The second customization point is conceptually simple. It can be done by enumerating the mounted modules inside one readout unit and give each module a unique name. For simplicity, this is done using a domain specific language (DSL). Listing 2.1 shows an example of how the composition of unpackers is done in the Elder-DSL.

¹ Unpacking is the process of extracting amplitudes and other recorded information from the raw data word bit-patterns written by the data acquisition system.

² in nuclear structure physics the VME standard is mostly used

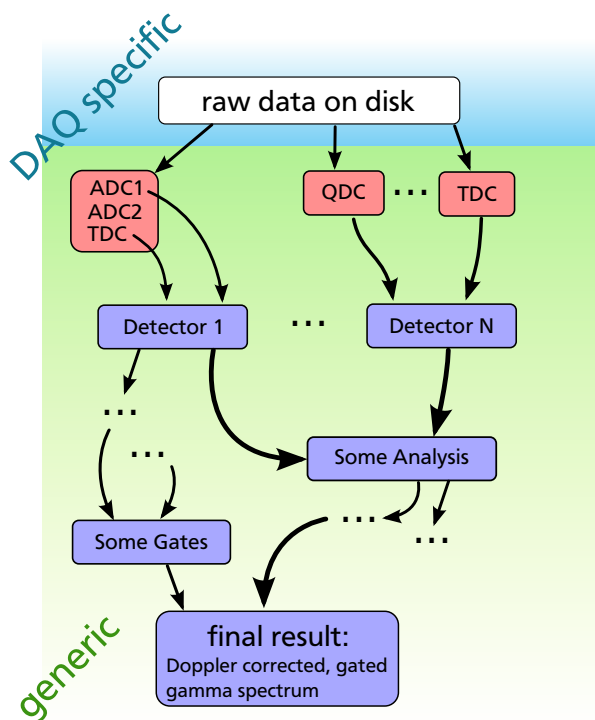


Figure 2.1.: Schematic representation of necessary steps to analyze a single event. Raw data is grouped in different readout units (red boxes), containing raw data of different physical acquisition modules (ADC, TDC and QDC). Unpacked data is copied to data processing nodes (blue boxes). The directional and acyclic nature of the graph (green background) is obvious, and represents a generic structure of all analysis programs. It is also indicated that the access of the raw data source is not generic (blue background), since the data format cannot be known by the framework developer. It could also come from different sources, such as files on a hard drive or it could be streamed over the network.

Data processing and reduction

Once the raw data is converted into meaningful numbers, such as signal amplitudes or time differences, the following steps of the data analysis process are commonly referred to as data reduction. This can be for example: calibration of ADC values, the calculation of position information from a particle tracking detector, or the Doppler correction of recorded γ -rays based on the measured particle velocity and the γ -ray emission angle. Many of these procedures are very well encapsulated in the sense that they require a given amount of input data to perform a calculation and produce a well defined set of output data. Output data of one algorithm serve as input data to other procedures. Such a structure can be described as a directed graph with data processing and reduction at its nodes and data flow along its edges. The graph has to be acyclic to guarantee a finite analysis time. A schematic drawing of the analysis process is shown in figure 2.1.

Data flow

It is an important design goal, that the development of data reduction algorithms is independent of the particular environment where the algorithm is being used. The motivation for this is a reduction of work that is needed to integrate new algorithms into existing data analysis software of a hosting experimental facility. It was observed in practice that it is time consuming and error prone to integrate external algorithms into the software system of the hosting facility if the interface for data exchange is different in the algorithm and the hosting program. The added conversion layers³ decrease the readability and maintainability of the program. As experiments are growing in size and complexity this problem will only become larger. Moreover, most detectors are not stationary after development and are used at different facilities, where the integration work has to be done again. An ideal situation would be, if all facilities and all detector developers would agree on a common data exchange format and would use a system

³ usually written in a rush, just before the beam-time starts

that enforces compliance to that format. The Elder framework developed in this work shows one possible way to realize such a standard.

In the context of data exchange on a directed graph, this goal can be achieved by using a limited set of high-level data types for the information flow between data processing nodes. Any finite set of data types is limiting the possibilities of input and output data layout. A trade-off has to be made between simplicity of the framework development on the one hand, favoring fewer different data types, and the possibilities available to algorithm developers on the other hand, favoring more different data types. Other aspect of this trade-off is the running speed of the final analysis program: a very flexible data type can perform slower in situations where a simpler data type would be sufficient. In the actual design, two different data types are provided that will be discussed in detail in the following section. The task of describing the data flow between the nodes of the directed graph is conceptually very simple. That is why the graph layout was integrated in the same DSL that is already used to compose unpacking modules for the raw data. Listing 2.1 shows how unpacked data from a crate is copied into a processing node in the Elder-DSL.

Data visualization

Data visualization is identified as another generic task of event-based data analysis. It is important for the diagnosis of experimental setups to look at histograms of particular quantities in the data flow of the analysis graph. The framework should allow the creation of a histogram by simply mentioning the value that should be filled into the histogram. Everything else, such as the range or binning of the histogram, can be done automatically. Creation of histograms is integrated into the same DSL that is used for unpacking and data flow definition. Listing 2.1 shows the creation of a x-y correlation histogram and an energy histogram of a double-sided silicon strip detector.

A very similar generic visualization system was already realized in the Cracow program [25] written by Jerzy Grębosz. However, Cracow does not offer a customization of data unpacking or data processing.

Closely connected to the task of visualization is the task of conditional analysis. Often, some quantity inside the data flow is decisive for the treatment of other data. As an example, the outgoing particle identification detector system calculates the charge of the outgoing particle after passing the reaction target. Based on this number, the type of nuclear reaction can be determined and the treatment of the data from γ -detectors might depend on that. Usually, these decisions are done based on one or two dimensional ranges that have to be displayed as an overlay on top of one or two dimensional spectra of the value that is tested against that range. The need to define these one or two dimensional conditions on histograms is so common, that it should be part of the framework. Many tools for visualizing histograms and conditions are available. In order to respect user preferences, the actual way of visualizing should not be fixed to a certain program or library, even if the creation of histograms and their association to a condition should be handled by the framework. To achieve this, the actual visualization tool is a customization point. The framework provides a C++ application programming interface (API) for this purpose. Most visualization tools allow to be adapted for different experiments, and any such tool⁴ can make use of the Elder framework functionality by calling Elder-API functions.

⁴ if the adaptation is done in C++

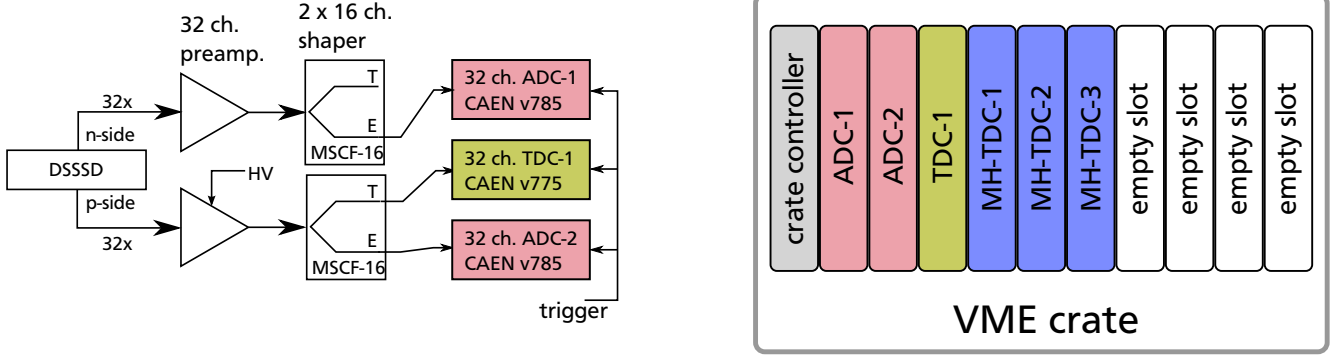


Figure 2.2.: A simplified subset of the PreSPEC setup, which is used to present the Elder features. A schematic drawing of the LYCCA target DSSSD readout and data acquisition is shown on the left hand side. The detector provides 32 signals on each side, which are pre-amplified before being processed by a shaper/CFD. On the right hand side, a schematic drawing of the physical arrangement of the DAQ modules inside a VME crate can be seen. This points out the modularity of the readout electronics, and the similarity of the physical crate and the crate block in the Elder-DSL script in listing 2.1.

2.2 Design of the software framework

Based on the previous section, a software framework was designed and written in C++. In this chapter design decisions will be discussed and the functionality will be presented. The main features are introduced by means of examples. The framework is called Elder and is publicly available [24]. It is intended to be used by other C++ programs. The software framework contains three main components:

- A parser for the Elder-DSL Grammar that reads Elder-DSL scripts, such as listing 2.1, and build the corresponding analysis graph.
- An API that makes Elder functionality available to other programs.
- A set of general data analysis algorithms.

2.2.1 Data flow description with the Elder-DSL

As presented in the introduction 2.1, three of four groups of generic tasks are only descriptive and do not involve complex algorithms: the arrangement of modules in crates, the data flow in the analysis graph, and the list of data that has to be visualized in histograms. Instead of using the general purpose language C++ for these types of user customization, a much simpler domain specific language (DSL), the Elder-DSL, was defined with simplicity and user friendliness as the main goal. As a simple example, the LYCCA target DSSSD readout is presented⁵. This example will give an overview of how the Elder framework is used in practice. The DSSSD setup is shown in figure 2.2. The complete analysis topology can be specified in an Elder-DSL script as shown in listing 2.1.

```
1 using LYCCA # Load a shared library with name libelderptLYCCA.so .
```

⁵ see section 4.1 for a description of the LYCCA detector system

```

2
3 crate TargetTofCrate      # Specify a DAQ readout unit and name it: TargetTofCrate.
4   procid 1                # A number to identify the correct part of the raw data.
5   module adc1 LYCCA.ADC    # Sequentially unpack data of two ADCs, using the ADC
6   module adc2 LYCCA.ADC    #   unpacker defined in the LYCCA shared library.
7   module tdc1 LYCCA.TDC    # The module keyword is followed by the module name and
8                             #   and the unpacker type.
9                             # Missing modules are ignored. The three multi-hit TDCs
10                            #   will not be unpacked.
11 end                       # finalize the readout unit with the 'end' keyword
12
13 # The first processing node for data reduction: The DSSSD analysis algorithm
14 #   is defined inside the LYCCA shared library as a class with name 'Dsssd'.
15 processor TargetDSSSD LYCCA.Dsssd      # Node name is "TargetDSSSD".
16   p_amplitudes[0:31] <- TargetTofCrate.adc1[0:31] # Take the ADC raw values
17                                                    #   (p-side)
18   n_amplitudes[0:31] <- TargetTofCrate.adc2[0:31] # and from the (n-side).
19                                                    # Note the simultaneous
20                                                    #   assignment of 32
21                                                    #   values in a single line.
22   histogram x:y           # Define a 2D-histogram and fill it with the particle coordinates.
23   histogram dE            # Define a 1D-histogram and fill it with energy loss information.
24 end                      # Finalize the processing node with the 'end' keyword.

```

Listing 2.1: Elder-DSL example, showing the unpacking and processing of data of the PreSPEC target DSSSD detector. The analysis topology is described by a graph with two nodes. One unpacking unit (crate) and one processing node (processor). The description of the sequential unpacking is as modular as the electronics (see figure 2.2).

Nodes of the analysis graph are defined by two building blocks: **crates** and **processors**. In listing 2.1, there is one node of each kind. Each block starts by the corresponding keyword, **crate** or **processor**, followed by a unique node name, and each block ends with the **end** keyword. For the processor block, the unique name of an algorithm must follow. Algorithms for LYCCA analysis are provided by shared libraries filenames such as `libelderptLYCCA.so` in listing 2.1 for the library with name `LYCCA`. Libraries are activated inside an Elder-DSL script by **using** them.

For the selection of the subset of raw data that is passed to a **crate** node, the **procid** keyword can be used. Sometimes, not all DAQ modules are producing data for all trigger conditions. In such cases, the **module** line can be extended by listing the trigger numbers when it should be unpacked. For example,

```
module adc1 LYCCA.ADC triggers 1 2
```

would only take action in an event where the trigger condition was 1 or 2.

Information flow between nodes is given by the edges in the graph. These are defined in the receiving node, which is the `TargetDSSSD` in listing 2.1. The algorithm specifies what information it needs as inputs in order to work. In the case of the `LYCCA.Dsssd` processor, the information is the amplitude of the p-side and n-side strip of the detector. The information about input and output values of any processor must be part of the processor library documentation

(libLYCCA.so in this case). The connection between the node `TargetTofCrate.adc1` and the `TargetDSSSD.p_amplitudes` is established in line 16 of listing 2.1 with the assignment symbol `<-`, an arrow that points in the direction of data flow. The DSSSD detector is read by two 32-channel ADCs, and the cabling was systematically done, such that strip number 0,1,2,... of the detector was plugged into channel 0,1,2,... of the ADC.

The Elder-DSL script exploits the modular nature of raw data unpacking as far as possible. Consider the setup in figure 2.2, but an additional DSSSD detector is added. The detectors are named A and B, respectively. The DAQ-team mounts two more ADC modules and one more TDC module into the crate. In order to adapt the analysis software for the changes in hardware, the Elder-DSL script has to be updated from listing 2.1 to the following:

```

1 using LYCCA
2
3 crate DsssdCrate
4 procid 1
5 for $DET in A B          # Modules for the DSSSD detectors A and B.
6   module adc1_$DET LYCCA.ADC # The suffix $DET is replaced with A and B
7   module adc2_$DET LYCCA.ADC # within the for block
8   module tdc1_$DET LYCCA.TDC
9 end
10 end
11
12 # Create two processing nodes for DSSSD detectors. Give them the suffix A and B
13 for $DET in A B
14   processor TargetDSSSD_$DET LYCCA.Dsssd # Node names are "TargetDSSSD_A"
15                                           # and "TargetDSSSD_B".
16   p_amplitudes[0:31] <- DsssdCrate.adc1_$DET[0:31]
17   n_amplitudes[0:31] <- DsssdCrate.adc2_$DET[0:31]
18
19   histogram x:y      # Define a 2D—histogram and fill it with the particle coordinates.
20   histogram dE       # Define a 1D—histogram and fill it with energy loss information.
21 end                 # Finalize the processing node with the 'end' keyword.
22 end

```

Listing 2.2: The Elder-DSL script after updating the experimental setup to have two DSSSD detectors.

The readout structure of the experiment becomes obvious when written in that way, much more than in a lengthy lookup table, possibly hard-coded inside the analysis program, where all modules are written explicitly. If there are cabling mistakes, or intended exceptions to the systematic cabling, these will be obvious to the reader of the Elder-DSL script, because the exceptions are treated as such in the code and have to be written outside of the `for` block. An Elder-DSL script is a functional summary of the signal routing and hardware setup for an experiment that can be read and understood by anyone without much effort. It facilitates development and maintenance compared to the development and maintenance of an analysis program written entirely in C++, or any other general purpose language, where all kinds of problems unrelated to the

data flow and unpacker composition can occur, and where the structure of the code may vary depending on the personal style of the developer.

Data types on the graph

As mentioned in the introduction of this chapter, high level data types are needed to achieve complete independence of the analysis algorithms. Any fixed set of available data types is a compromise between development effort for the framework development, efficiency in the execution and flexibility in the algorithm development. In any case, the available types must be expressive enough to handle any practically relevant case of data transfer. That means, at least one complex data type must be there to handle any thinkable situation. Simpler types can be added to increase efficiency and user friendliness for particular situations. The Elder framework in its current state contains two data types, a complex and a simple one.

Single values

The simplest, yet relevant piece of information that can move along the graph edges is a single floating point number. Each such value has a globally unique name as a *static*⁶ property. Such a number is produced by one node in the graph, and copied along the graph edges to one or more nodes that need this specific number as input. In some cases, a computation cannot be reasonably done if an input value is missing. For that reason, each value has a Boolean flag that contains information about the validity of the number. If an algorithm produces a number as output, its valid-flag is automatically set by the framework. In all other cases, this flag remains invalid and all following algorithms in the data flow path can check that flag to prevent doing computations with invalid input data. Note that in many current analysis software systems, magic numbers⁷ are used for this purpose. This is bad practice, because the magic number can fall inside the range of valid input data of a data processing stage. It may also happen, that at one place the check for the magic number is forgotten, or a variable is accidentally not initialized with the correct value. If such a value propagates a calibration step, it might not be recognizable as the default value anymore and produce artifacts in the result.

A histogram of a single value can be created inside the Elder-DSL script by writing **histogram x** inside a **processor** block. For each event, where the valid flag of **x** is true, the value of **x** is sorted into the corresponding histogram. A two dimensional histogram, containing a correlation plot of two valid single values **x** and **y**, can be created by writing **histogram x:y** inside a **processor** block. In this case, for each event the tuple (**x,y**) is sorted into the histogram if both, **x** and **y** are valid. Single value assignment is done using the arrow **<-** operator, which copies the content of the right hand side to the left hand side.

Elder arrays

Single values are only theoretically sufficient to do all possible analysis tasks. For a larger number of similar signals, single values are very inconvenient. The Elder framework provides a second data type called *array*. Each array has a unique static name. Elder arrays are lists of pairs of a floating point *value* and an integer *index*, where the index is limited to a range from zero to $N - 1$. That makes N another static property of an array. Arrays can be inputs

⁶ the same for each event

⁷ typically large positive or negative numbers

name=A	entry#	0	1	2	3	4	5	name=B	entry#	0	1	2	3
N=10	index	5	1	1	0	4	9	N=10	index	5	1	1	0
length = 6	value	3.2	1.5	0.9	-0.2	6.6	4.4	length = 4	value	3.2	1.5	0.9	-0.2

name=empty	entry#							name=C	entry#	0	1	2	3	4	5
N=10	index							N=1	index	0	0	0	0	0	0
length = 0	value							length = 6	value	3.2	1.5	0.9	-0.2	6.6	4.4

Figure 2.3.: Four examples of valid arrays with different static names A,B,empty,C, index ranges 10,10,10,1, and different dynamic lengths 6,4,0,6, respectively. In the array with name C, the index is useless. Another data type without index would be more appropriate in this case. But such a data type is not implemented in Elder yet.

or outputs of algorithms and their static properties, name, and index range N , are part of the processor implementation. The array length, i.e. the number of (value, index) tuples is a *dynamic*⁸ property of the array. The array length has no upper limit and multiple entries inside the array may have the same index. All entries in an array are valid, therefore, array entries do not need a valid flag. Examples of valid arrays and their content are given in figure 2.3. In practice, arrays may contain data from a simple ADC module with multiple channels, where the index would refer to the ADC channel number and the value would correspond to the measured amplitude in that channel. Because the same index can occur multiple times in an array, it can store data from more complicated modules such as a multihit-TDC module. Another use case is to handle data from a γ -ray detector array, where the index would refer to the detector number and the value represents the measured amplitude in that detector.

Visualization of arrays is more complicated than for single values. Consider an array with name `array` and index range N . Inside a `processor` block in an Elder-DSL script, the syntax `histogram array` causes the creation of N histograms, indexed from 0 to $N-1$. For each event, all (value, index) pairs of the `array` are sorted into the histograms by putting the value into the histogram addressed by index. In order to sort all values into one histogram, independent of their index, a helper processor must be used. This processor is part of the `libelderstd.so` library as part of the Elder software package. Listing 2.3 shows how to do this using a `for` block.

Correlation histograms of arrays with names `array1` with length l_1 and `array2` with length l_2 can be created in different ways. This is already pointed out in a systematic way by Jerzy Grębosz in [25]. In Elder, there are four different possible meanings of the correlation of two arrays: blind pairs, pairs, combinations and all combinations. The meaning of this is explained in figure 2.4. They correspond to different treatment of the entries based on their indices. Blue arrows represent the tuples of values, that are sorted into the 2D histogram.

- The first way of creating tuples is called *blind pairs*, and the Elder-DSL syntax for this is `histogram array1:array2`. In this case, indices are ignored. The first tuple is created by the first entry of `array1` and the first entry of `array2`. The second tuple is created by the second entry of both arrays, and so on. For each event, there will be $\min(l_1, l_2)$ tuples created and filled into the histogram. Using blind pairs makes sense, only if both arrays

⁸ may be different for each event

```

1 processor Proc Library.ProcessorA
2   # Provides an array output: array (length = 3).
3   histogram array # create 3 histograms
4 end
5 using std
6 processor Helper std.array
7   for $i in [0:2]
8     entry[0] <- Proc.array[$i]
9   end
10  # the for block will be expanded to
11  # entry[0] <- Proc.array[0]
12  # entry[0] <- Proc.array[1]
13  # entry[0] <- Proc.array[2]
14  histogram entry # creates a single histogram
15 end

```

Listing 2.3: Workaround to create a single histogram of an array with different indices.

are intentionally created without meaningful indices. This way would be more natural for a data type without indices⁹, which does not exist in the current version of Elder.

- The second way is called *pairs*, and builds the tuples by finding all pairs of entries with the same index. The Elder-DSL syntax for this form of array correlation is **histogram array1::array2**, with two colons. An application of this kind of correlation is for example the energy-time correlation of a detector array. In that case, the index would correspond to detector number. If energy information is contained inside an array **energy**, time information is contained in another array **time**, and the indices of both arrays consistently refer to the detector number, the energy-time correlation plot can be created by writing **histogram energy::time**.
- The third way is called *combinations*, and builds the tuples by taking all pairs of entries with different indices. A use case for this is the creation of a γ - γ -coincidence matrix for a γ detector array. For a symmetric matrix, all detectors¹⁰ are contained in a single array **gamma_array**. The matrix can be created by writing **histogram gamma_array::gamma_array**. In case of a non symmetric matrix, different detector groups must be assigned to different arrays, e.g. **gamma_groupA** and **gamma_groupB**. The matrix will then be created by writing **histogram gamma_groupA::gamma_groupB** or **histogram gamma_groupB::gamma_groupA**. The resulting histograms are transposed, but otherwise identical.
- The fourth way is called *all combinations*, and builds tuples from all possible pairs, ignoring indices. This can be useful if two detectors measure the same quantity, for example the x-position of a particle. If both detectors have multi-hit capability, the correlation of both x positions can be seen by writing **histogram xpos1:::xpos2**, with four colons. This

⁹ see the definition of multi-values below

¹⁰ including all indices

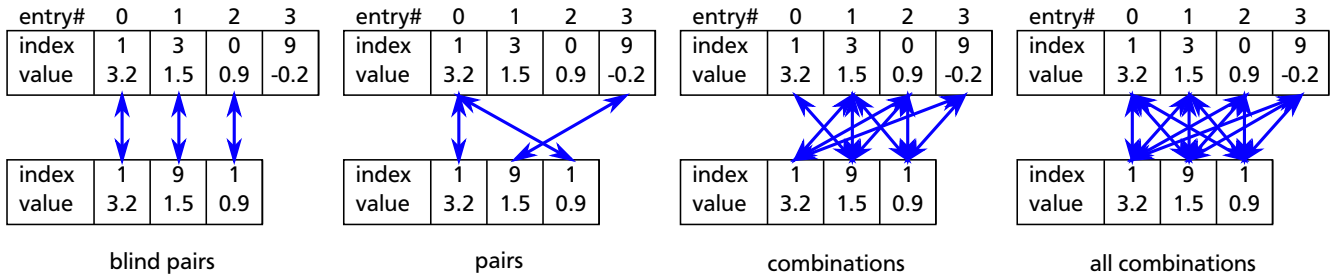


Figure 2.4.: In any case, creating a 2D histogram from two arrays requires the formation of value tuples that are sorted into the histogram. There are four different meanings of doing this with Elder arrays. In each of these cases, blue arrows form tuples from the values in the array entries they point to.

mode of correlation would be more natural for an array type without indices, which does not exist in the current version of Elder.

Multi-values

This data type does not exist in Elder, but would be a useful future addition. It could be used in all cases where arrays are used without referring to the indices, for example in correlation histograms using *blind pairs* and *all combinations*. The multi-value type would be identical to the array type, but without the index field. Naturally, the syntax `histogram mv` would create one histogram that will get all entries of the multi-value `mv`. In case of correlation histograms, there would be two modes, *pairs* and *combinations*, that would be equivalent to the *blind pairs* and *all combinations* of the array type. At the same time, this would as well reduce the correlation modes for arrays to the two modes, *pairs* and *combinations*, while *blind pairs* and *all combinations* could be removed.

Assignment

Given the two data types, arrays and single values, the rules of data assignment have to be defined. Assignments are possible between both types in both directions. The rules are designed to be intuitive. Their definition is given by the enumeration of all possibilities in listing 2.4. Arrays can contain multiple entries with the same index. In case of an assignment like `val <- A.array[2]`, the value of `val` will be equal to the last value with index 2 in `A.array`. Information can be lost in such an assignment. If a specific hit selection of an array has to be done, the selection scheme should be implemented inside a `processor` as part of a custom library that is experiment specific, as described in the next paragraph.

```

1 processor A Library.ProcessorA
2   # Provides a single value output: value1.
3   # Provides a single value output: value2.
4   # Provides an array output: array (index range N = 5).
5 end
6
7 processor B Library.ProcessorB
8   # Has a single value input: val.
9   # Has an array input: arr (index range N = 4).
10

```

```

11 # Assign single value to single value.
12 val <- A.value1
13 # Assign single value to array index.
14 arr[1] <- A.value1 # insert an entry into array with index 1 and value equal to A.value1.
15 arr[2] <- A.value1 # insert an entry into array with index 2 and value equal to A.value1.
16 arr[3] <- A.value2 # insert an entry into array with index 3 and value equal to A.value2.
17 # Assign array index to array index.
18 arr[1] <- A.array[0] # insert all entries of A.array with index 0 into array
19                        # and change index to 1.
20 arr[2] <- A.array[1] # insert all entries of A.array with index 1 into array
21                        # and change index to 1.
22
23 arr[1:2] <- A.array[0:1] # Assign multiple indices sequentially,
24                        # equivalent to the two previous lines.
25 arr[1:2] <- A.array[1:0] # Same as above, but reversed:
26                        # arr[1] <- A.array[1]
27                        # arr[2] <- A.array[0]
28 # Invalid assignment trigger an error condition
29 arr[0:3] <- A.array[0:4] # ERROR: inconsistent index range
30 arr[10] <- A.value1      # ERROR: allowed indices are only [0],[1],[2], or[3]
31 # Assign array indices to single values.
32 # In case of multiple entries with index 2, the last value is taken.
33 val <- A.array[2] # Take last entry with index=2, and copy it's value into val.
34                  # Note that in case of multiple entries with index=2, information
35                  # is lost.
36 end

```

Listing 2.4: Assignment rules for Elder data types. In all cases, Elder-DSL will check if the correct number of channels were assigned (lines 25, 26 in listing 2.4) and signal an error to the user if the index ranges are inconsistent.

2.2.2 Adding analysis algorithms

Until now, it was assumed that the analysis algorithms already exist, and were used as building blocks of the analysis. This section introduces the basic mechanism of defining new processors. It also explains the relation between the names of inputs and outputs (single values and arrays) in the Elder-DSL script and the corresponding values in the C++ code of the algorithm implementation. This mechanism is what makes true modular development possible.

Class declaration

All algorithms are represented by C++ classes that inherit from the `elder::process::Processor` base class, provided as part of the Elder-API. Listing 2.5 shows the C++ header file for a simplified version of an analysis algorithm for a double sided silicon strip detector (DSSSD) with 32 strips on each side. This processor was used in the Elder-DSL script in listing 2.1. The purpose of this example is to present some of the main features of the Elder API and how to write algorithms that can be used in Elder-DSL scripts.


```

1 #ifndef DSSSD_HPP_
2 #define DSSSD_HPP_
3 #include <elderpt.hpp>
4 class DSSSD : public elderpt::process::Processor {
5 public:
6     // The processor initialization .
7     virtual void initialize();
8     // The process method where the algorithm is implemented.
9     virtual void process();
10    // The interface documentation is part of the class declaration.
11    enum InputArray { // list all input arrays to this processor as C++ enumeration
12        amplitude_p, // adc amplitudes of p-side channels.
13        amplitude_n, // adc amplitudes of n-side channels.
14    };
15    enum Output {
16        x,           // x-position
17        y,           // y-position
18        dE,          // energy loss
19    }
20    enum Coefficients {
21        cal_p,       // calibration coefficients for p-side channels.
22        cal_n,       // calibration coefficients for n-side channels.
23    }
24 };
25 #endif

```

Listing 2.5: A very simple processor for DSSSD analysis.

The enumerations are used to list all input and output data of the algorithm, and at the same time, establish a link between C++ data and the names to access it in the Elder-DSL script. That makes the header of an Elder **processor** file a useful piece of documentation by itself. In addition to the **enum** values, a processor class must provide an **initialization** and a **process** method.

Processor initialization

Inside the **initialization** method, the **enum** values are passed as arguments to Elder macros, to notify the base class **elderpt::process::Processor** of the inputs and outputs and their names, as can be seen in lines 6-14 of listing 2.6. After all **enum** values are treated, the **init()** method has to be called. If parameters, conditions or calibration coefficients were defined, the corresponding methods (in this case only **read_coefficients()**) have to be called.

```

1 void DSSSD::initialize()
2 {
3     // specify inputs
4     NAME_INPUT_ARRAY(amplitude_p, 32); // fix the index range to 32
5     NAME_INPUT_ARRAY(amplitude_n, 32);
6     // specify outputs

```

```

7  NAME_OUTPUT_CHANNEL(x);
8  NAME_OUTPUT_CHANNEL(y);
9  NAME_OUTPUT_CHANNEL(dE);
10 // specify calibration coefficients
11 NAME_COEFFICIENTS_ARRAY(cal_p, 32);
12 NAME_COEFFICIENTS_ARRAY(cal_n, 32);
13 // read calibration coefficients from file
14 read_coefficients();
15 }

```

Listing 2.6: Constructor implementation of a very simple processor for DSSSD detectors.

There are also macros `NAME_OUTPUT_ARRAY(name, N)` and `NAME_INUPT_CHANNEL(name)`, for the definition of output arrays and input single values. Both are not used in this example. These macros will provide the enumerated values as interface to the Elder graph framework, such that in an Elder-DSL script an assignment `amplitude_p[0:31] <- DsssdCrate.adc1[0:31]` is possible. Calling `read_coefficients()` opens a file, that should contain the calibration coefficients for this algorithm as defined by the macro `NAME_COEFFICIENTS_ARRAY(name, N)`. If no such file exists, it is created containing default values. Comments in that file can be introduced with the hash character `#`, and go until the end of the line. Each line starts with a character string that specifies the name of the coefficient, followed by a list of floating point numbers. These numbers are interpreted as coefficients of a polynomial. The first number is the coefficient for order zero, the second number for the coefficient of order one, and so on. If the coefficient was defined using the array macro `NAME_COEFFICIENTS_ARRAY(cal_n, 32)`, there will be 32 lines containing the same coefficient name and an index in square brackets like this:

```

1 cal_p[0]  1.6  1.05
2 cal_p[1]  0.4  1.31
3 # ...
4 cal_p[31] -0.1  1.12
5
6 cal_n[0]  1.4  0.92
7 # ...
8 cal_n[31] 0.1  0.87

```

Listing 2.7: A file with calibration coefficients for DSSSD calibration. Here, two numbers per coefficient are given, i.e. a linear calibration is used. Higher order polynomial calibration can be used by writing more numbers in the list of coefficients.

The file name of this calibration file is derived from the name of the processor node inside the Elder-DSL, with the `.cal` extension. The forward slash `/` character is allowed inside processor names, to organize the calibration files in directories. If the calibration file with the coefficients does not exist, it is created with default values `0 1` for all coefficients. Besides calibration coefficients, Elder-DSL processors can have two more data files with parameters for algorithms: conditions `.con` and parameters `.par`. They are used in the same way as calibration coefficients, only that their initialization inside the C++ class constructor is done with `NAME_CONDITION_ARRAY(condition_name, 32)`, `NAME_PARAMETER_ARRAY(parameter_name, 32)`, `read_conditions()`, and `read_parameters()`, respectively. There are versions of the macros without the `_ARRAY` extension. In this case there are no indexing square brackets after the name in the corresponding

file. The framework does all book-keeping, such as memory allocation and file handling, so that the user can focus on implementing an algorithm, and only has to specify his needs in terms of input, output and parameters. In addition, any algorithm implemented in this way can be used in any context inside an Elder-DSL script without making any modifications to C++ code. That is what is meant by truly modular and reusable design.

Algorithm implementation

The starting point of any algorithm defined using the Elder-API is the `process`-method inside the processor class. When that method is called within the Elder framework, all parameters are already available and the input data from the current event is present. It remains to explain how to access the input data, as well as the parameters/coefficients/conditions, and how to provide the computed results as output of the processor node to the framework which will copy it to all successive processors in the analysis graph.

```
1 // implementation of the algorithm
2 void DSSSD::process() {
3     // loop over all p-side entries and get index and value
4     double xpos;
5     double ypos;
6     double energy_loss;
7     for (int i = 0; i < input_array_size(amplitude_p); ++i) {
8         // This is the way to
9         int index = input_array_index(amplitude_p, i); // loop over any array:
10        double value = input_array_value(amplitude_p, i); // extract index and value
11        // based on the entry index i.
12        double energy = calibrate(cal_p, value, index);
13        // ... determine xpos
14    }
15    for (int i = 0; i < input_array_size(amplitude_n); ++i) {
16        // do the same for n-side and ypos
17        // ...
18    }
19    // fill the outputs of the processing node
20    set_output(x, xpos);
21    set_output(y, ypos);
22    set_output(dE, energy_loss);
23 }
```

Listing 2.8: Reading the input and providing the output for the simple DSSSD processor. Note the way of looping over the array `amplitude_p`. The for-loop variable `i` enumerates all array entries, and the number of entries is obtained by calling the method `input_array_size(amplitude_p)`. The `index` and `value` can then be obtained from the `i`-th array entry.

The valid flag for the output values is automatically set to true if the method `set_output(output_name, value)` is called. If single values for input are used, the accessing method would be `input_valid(single_value_name)` for the valid flag and the method `input_value(single_value_name)` for its value. To access output arrays, the method with name

`fill_output_array(array_name,index,value)` can be used. Table 2.1 summarizes the usage of input/output data and parameters inside processors.

2.2.3 Adding module unpackers

As described in section 2.2.1, two types of nodes are present in the data flow graph: processors and crates. Processors contain algorithms for data reduction and crates are the source of the data flow. The concept of crate-nodes is inspired by the modular nature of data acquisition with modules in (VME-) crates. This is so common, that it is a core concept in the Elder-DSL to combine modules in crates, using the Elder-DSL keywords `crate` and `module` (see listing 2.1). The Elder API provides an interface for raw data unpackers. Like the interface for processors, unpackers have to be C++ classes that inherit from the `elder::unpack::Unpacker` class, which is part of the Elder-API. Every raw data unpacking code written as a C++ class using the Elder-API can be used as `module` in an Elder-DSL script.

Inside each unpacker class, the method `unpack_data(const uint32_t *data_ptr, const uint32_t *data_end)` has to be implemented. When the analysis is running, the Elder framework will select from the raw data all readout units that match a given `crate` block. It will then pass a pointer to the first word of the raw data from the matching readout unit as first argument to the `unpack_data` method of the first unpacker module inside the `crate` block. The second argument to that method is a pointer to a position right after the last valid raw data word of the matching readout unit. This should be used by the `unpack_data()` method to prevent reading beyond the valid data range. The return value of `unpack_data()` is a pointer after the last 32-bit word that was read. This pointer is passed as first argument to the `unpack_data()` method of the next unpacker module in the crate block. In this way, all unpacker modules are processing the raw data in a sequential way. In addition to the `unpack_data()` method, the `clear()` method has to be implemented. This should reset the internal state of the class¹¹. The `clear()` method is called by the framework before `unpack_data()` is called. In addition, there are six more methods to implement. Two of them are used to provide some static data about the unpacker to the Elder framework. These are:

- `int num_channels()` returns the number of different signal channels of this module. Occasionally modules produce multiple information per channel, e.g. energy and time. In this case, the unpacker can provide the energy and time information at different channel indices. For example, a module can provide energy and time information for each hit in each of its 8 channels. The unpacker can then provide 16 output channels, where channels 0 to 7 will contain energy information and channels 8 to 15 will contain time information.
- `int raw_range()` returns an integer that defines the number of bins in the raw spectrum created by the Elder framework if raw spectra are enabled for a crate.
- `bool jitter()` returns a boolean that determines if a random jitter is added to each value (see below). This method has a default implementation that returns `true`.

There are three methods for data access. Data access to unpackers is similar to Elder arrays. After unpacking is done, each unpacker has a list of hits of (channel, value) pairs. The pairs correspond to the (index, value) pair in Elder arrays. The different naming convention is used,

¹¹ If the class has no internal state, this method can be empty

Type	Definition (.hpp)	Initialization (.cpp)	Usage
single value input	enum Input{my_input,};	NAME_INPUT_CHANNEL(my_input); init();	bool valid=input_valid(my_input); double value=input_value(my_input);
input array	enum InputArray{my_iarray,};	NAME_INPUT_ARRAY(my_iarray, index_range); init();	int size=input_array_size(my_iarray); int index=input_array_index(my_iarray); // ∈ [0, index_range[double value=input_array_value(my_iarray);
single value output	enum Input{my_output,};	NAME_INPUT_CHANNEL(my_output); init();	double ovalue; value=set_output(my_output, ovalue); bool valid=output_valid(my_output); double value=output_value(my_output);
output array	enum OutputArray{my_oarray,};	NAME_OUTPUT_ARRAY(my_oarray, index_range); init();	int oindex; // ∈ [0, index_range[double ovalue; fill_output_array(my_oarray, index, value);
parameter	enum Parameter{my_par,};	NAME_PARAMETER(my_par); init(); read_parameters();	double par0=parameter(my_par); // or int N = parameter_count(my_par); for (int n=0; n<N; ++n) double par=parameter(my_par, n);
parameter array	enum Parameter{my_parray,};	NAME_PARAMETER_ARRAY(my_parray, length); init(); read_parameters();	for (int index=0; index<length; ++index) { double par=parameter_array(my_parray, index) // or int N = parameter_count(my_parray, index); for (int n=0; n<N; ++n) double par=parameter_array(my_parray, index, n); }
condition	enum Condition{my_con,};	NAME_CONDITION(my_con, T); // T = Window1D or Window2D or Polygon2D; init(); read_conditions();	double x, y; condition_valid(x, my_con); // Window1D condition_valid(x, y, my_con); // Window2D or Polygon2D
condition array	enum Condition{my_carray,};	NAME_CONDITION_ARRAY(my_carray, T, length); // T = Window1D or Window2D or Polygon2D; init(); read_conditions();	for (int index=0; index<length; ++index) { double x, y; condition_valid(x, my_carray, index); // Window1D condition_valid(x, y, my_carray, index); // Window2D or Polygon2D }
coefficients	enum Coefficients{my_coeff,};	NAME_COEFFICIENTS(my_coeff); init(); read_coefficients();	double raw; double cal=calibrate(raw, my_coeff); // or double cal=calibrate_at_offset(raw, my_carray);
coefficients array	enum Coefficients{my_carray,};	NAME_COEFFICIENTS_ARRAY(my_carray, length); init(); read_coefficients();	for (int index=0; index<length; ++index) { double raw; double cal=calibrate(raw, my_carray, index); // or double cal=calibrate_at_offset(raw, my_carray, index); }

Table 2.1.: Summary of all data/parameter access methods for initialization and access. This table was published inside the PresPEC collaboration as part of the prespec software package documentation.

because it is more common to talk about channels of electronic modules, rather than indices of electronic modules. Another difference is, that values are not floating point numbers (**double**) but 32-bit unsigned integers, because electronic modules tend to output integer values.

- `size_t size()` returns the number of hits in the module.
- `int channel(size_t i)` returns the channel of the *i*-th hit.
- `uint32_t value(size_t i)` returns the value of the *i*-th hit. If the `jitter()` method (see above) returns true (which is the default implementation), the Elder framework will add a uniformly distributed random number in the range $[0, 1[$ to the returned value to avoid aliasing.

The definition of an unpacker class depends very much on the details of the raw data format. Therefore, no example will be given here, but common cases are contained as examples in the Elder software package [24].

2.2.4 The Elder API for analysis programs

The Elder package contains an API that can be used by another program to parse Elder-DSL scripts and use the Elder framework to analyze the data. The API consists of four classes: `elderpt::control::Controller`, `elderpt::control::Event`, `elderpt::control::Subevent`, and `elderpt::viscon::Interface`.

The `elderpt::control::Controller` is initialized with a `std::string` that contains the file-name of an Elder-DSL script. During initialization, the `elderpt::control::Controller` will create a data structure in memory that represents the analysis graph as described by that script. After initialization, the program can read raw data event by event¹², and analyze it using the Elder-DSL graph. The raw data has to be converted into a format that can be passed to the `elderpt::control::Controller` object. From that point, all analysis steps are handled by that object. The raw data format to which the program has to convert, is similar to the raw data format of the multi branch system (MBS), which is described in chapter 8 of [28], and represented by an object of type `elderpt::control::Event`. Each event contains information about the recording time and the trigger condition. This information can be given to the constructor of `elderpt::control::Event`. An event object contains a list of `elderpt::control::Subevents`, and each subevent holds the raw data of one read out unit, i.e. concatenated raw data from all modules inside that readout unit. Subevents are uniquely identified by a set of numbers that are called `procid`, `type`, `subtype`, `control`, and `subcrate`. The names for the identifying numbers were adopted from MBS data structures, and are used in Elder-DSL scripts inside the `crate` blocks to select the appropriate section of raw data. For example, in listing 2.1, the statement `procid 1` was used to refer to the DSSSD part of the raw data. Other DAQ systems might identify different sections of the raw data in other ways, and an appropriate mapping has to be implemented when filling the event object. The constructors of event and subevent classes are shown in listing 2.9.

```
1 elderpt::control::Event(uint32_t number,  
2                        uint32_t type,
```

¹² The reading of raw data is not part of the Elder framework, because the representation of the data format is completely up to the user.

```

3          uint32_t trigger,
4          uint32_t time,          // absolute time in seconds
5          uint32_t msec,         // millisecond fraction of time
6          uint64_t timestamp); // absolute time in best possible
7                                // resolution from the DAQ system
8
9 elderpt::control::Subevent(uint32_t procid,
10                            uint32_t type,
11                            uint32_t subtype,
12                            uint32_t control,
13                            uint32_t subcrate,
14                            uint32_t length,
15                            const uint32_t *data);

```

Listing 2.9: Constructors of event and subevent objects.

After the raw data and all related information have been copied into the event object, it can be passed to the `Controller` object by calling the three methods `control.clear()`, `control.unpack(myInterface, event)`, and `control.process(myInterface)` in that order. The first function will clear information from the previous event, the second will unpack the raw data, and the third will pass all data along all edges of the analysis graph.

The argument `myInterface` is an interface object for visualization of histograms and conditions. If the program has the possibility to create and display histograms, the Elder framework can make use of these possibilities by means of an interface class called `elderpt::viscon::Interface`. The namespace `elderpt::viscon` is an abbreviation for “visualization and conditions”. The client program has to define a class that inherits from `elderpt::viscon::Interface`, and can define the following 8 methods:

- `virtual int hist1d_create(const char *name, const char *title, const char *axis, int n_bins, double left, double right)`
- `virtual void hist1d_fill(Histogram1DHandle h, double value)`
- `virtual Histogram2DHandle hist2d_create(const char *name, const char *title, const char *axis1, int n_bins1, double left1, double right1, const char *axis2, int n_bins2, double left2, double right2)`
- `virtual int cond1d_create(const char *name, double left, double right, Histogram1DHandle h)`
- `virtual void cond1d_get(Condition1DHandle h, double &left, double &right)`
- `virtual void hist2d_fill(Histogram2DHandle h, double value1, double value2)`
- `virtual int cond2d_create(const char *name, const std::vector<double> &points, Histogram2DHandle h)`
- `virtual void cond2d_get(Condition2DHandle h, std::vector<double> &points)`

If any method is not defined, the respective functionality will be missing in the program, but the analysis is still able to run. For example, if only 1D-histograms are needed, it is sufficient to implement only the two methods related to 1D-histograms. If no visualization of gates is needed, the `condXX` methods do not need to be implemented. The `xx_create` methods return an integer value, that must be a unique identifier of the created histogram. That same integer is passed to the `xx_fill` methods, to identify the histogram that the framework wants to fill. Note that there is additional functionality implemented in terms of the one and two dimensional histograms. The `ratemeter` and `waveform` display depends on the availability of 1D histograms, and the `picture` display depends on the availability of 2D histograms.

The outline of a typical `main()` function of a front-end program can be seen in listing 2.10.

```

1 #include <elderpt.hpp>
2 #include <vector>
3 #include <MyHistogram.hpp>
4 class MyInterface : public elderpt::viscon::Interface
5 {
6     // a typical implementation
7     std::vector<MyHistogram> histograms;
8     virtual int hist1d_create(/* ... */) // see above
9     {
10         int handle = histograms.size();
11         histograms.push_back(MyHistogram(/* ... */));
12         return handle
13     }
14     virtual void hist1d_fill(Histogram1DHandle h, double value)
15     {
16         histograms[h].fill(value);
17     }
18 };
19
20 int main()
21 {
22     MyInterface myInterface;
23     elderpt::control::Controller
24         control("toplevel_analysis.config", myInterface);
25
26     // open raw data file
27     // ...
28
29     // loop over all raw data events
30     // and fill event structure
31     for (int e = 0; e < n_events; ++e)
32     {
33         elderpt::control::Event event(/* ... */); // see listing 2.9
34         for (int s = 0; s < n_subevents; ++s)
35         {

```



```

36     // get subevent raw data
37     // ...
38     elderpt::control::Subevent subevent(/* ... */); // see listing 2.9
39     event.push_back(subevent);
40 }
41
42 // pass event structure to Elder framework
43 control.clear();
44 control.unpack(myInterface, event); // all unpacking done
45 control.process(myInterface); // analysis done
46 }
47 }

```

Listing 2.10: Example of the outline of a front-end program using the Elder framework for data processing.

For further information, refer to the example implementations of `elderpt::viscon::Interface` classes in [24].

2.3 Conclusions

This chapter presented the key points of the Elder software framework design, which achieves the goals mentioned in the introduction: a robust implementation of reappearing task in the development of data analysis software, together with customization points for anything that really differs between different setups. Simple examples were used to describe how this framework can be used for data analysis with an Elder-DSL script, and how to write C++ programs using the Elder-API.

Based on some experience with the implementation I can make the statement, that any analysis system using the Elder framework is a lot easier to maintain, understand, extend and use, than typical programs that were written entirely in C++. Even if the implementation of a C++ program is modular, it is still easier to follow an Elder-DSL script than a C++ code in order to understand the structure of the data flow. The framework helps to develop new systems fast and with less possibilities to make mistakes, because existing components can be reused without any effort due to the high degree of modularity. Furthermore, the framework encourages the use of standardized parameter and calibration files.

Current usage

- During the PreSPEC campaign in 2014, the main tool for online analysis was a collection of tools that contained already the major features of the elder framework. What is published in [24] is the pure framework with slight improvements. The original PreSPEC analysis tools were distributed and are available inside the PreSPEC collaboration (see chapter A).
- The PreSPEC analysis tools were used to analyze all data presented in this work in the following chapters.
- It is currently in use at the university of Cologne to analyze data from the local MBS system [29].

-
- To test the flexibility of the API, a program was written in collaboration with M.L. Cortés, which allows to replay raw data from the REX-Isolde facility and allows to use Elder-DSL scripts for the analysis.

3 Nuclear collision dynamics and Coulomb excitation

In low-energy nuclear structure research, the majority of collision experiments is of the *fixed target type*, where a beam with known properties (energy, momentum, charge) is shot on a macroscopic piece of matter, typically a foil, and the collisions happen randomly with a certain cross section. Among this type of experiments, many use beam energies of below 10 MeV/u, corresponding to a velocity of $v/c < 0.145$, which can be considered as non-relativistic. In recent times, more experiments use energies of around a few hundred MeV/u, where $v/c \approx 0.5$ and relativistic effects become important. The reason for the high beam energy is the demand for experiments with beams of unstable, exotic nuclei which are often produced in reactions of a stable beam with a thick primary target and a subsequent selection of the desired isotope by a fragment separator. Higher energy is needed to maximize the production cross section of the desired isotope, to increase its transmission through the spectrometer with limited angular acceptance, and to allow thicker secondary reaction targets to compensate for the low beam intensity. These experiments are in the so called *intermediate energy* regime and are likely to be a major fraction of the upcoming nuclear structure experiments at the FAIR facility.

The aim of most of these experiments is to create inelastic reactions between the beam and target nuclei. This is useful to either study the reaction itself, or the subsequent decay of the nucleus to learn something about its structure. One important reaction mechanism is based on the mutual electromagnetic interaction of colliding nuclei and is called Coulomb excitation. In such a process, the distance between collision partners has to be larger than the range of the nuclear interaction to ensure a purely electromagnetic excitation. Because Coulomb excitation is caused by the time dependent fields of the collision partners along their trajectories, this chapter starts with a fully relativistic numerical simulation of these trajectories, similar to [30]. The resulting trajectories and fields are used to do a numerical coupled channel calculation of the excitation amplitudes of the nuclei without imposing any approximation. Common approximations are for example $v/c \ll 1$ in the so called classical Coulomb excitation [5] or the approximation of straight line trajectories at higher energies [31] which is often used in relativistic Coulomb excitation. The computer program developed in this chapter reproduces the results of both approximations and allows to calculate results for all intermediate situations.

3.1 Relativistic dynamics

In this section, the well-known equations of motion of charged particles in electromagnetic fields are applied to describe the process of two-body collisions.

The fundamental quantity in the special theory of relativity is the Lorentz-covariant four vector. For the solution of the problems discussed here, the relevant four vectors are:

$$\text{position:} \quad (x^\mu) = \begin{pmatrix} c t \\ \vec{x} \end{pmatrix} \quad (3.1)$$

$$\text{velocity:} \quad (u^\mu) = \frac{d}{d\tau} (x^\mu) = \gamma \frac{d}{dt} (x^\mu) = \gamma \begin{pmatrix} c \\ \vec{v} \end{pmatrix} \quad (3.2)$$

$$\text{momentum:} \quad (p^\mu) = \begin{pmatrix} E/c \\ \vec{p} \end{pmatrix} = m (u^\mu) = m \gamma \begin{pmatrix} c \\ \vec{v} \end{pmatrix} \quad (3.3)$$

$$\text{electromagnetic potential:} \quad (A^\mu) = \begin{pmatrix} \phi \\ \vec{A} \end{pmatrix}, \quad (3.4)$$

with the relativistic factor $\gamma = 1/(1 - (v/c)^2)^{1/2}$. The notation x^μ indicates the μ -th component of the four-vector, while (x^μ) represents the vector itself. For the electromagnetic quantities, the Gauss system is used because it simplifies the discussion of electromagnetic dynamics with four-vectors. In this system electric and magnetic fields have the same units, which seems natural as they emerge from the same four-vector potential. For the elementary charge e^2 the expression $\alpha \hbar c = 1.44 \text{ MeV fm}$ is used because it evaluates to the same number in SI and Gauss units. This allows to evaluate the formulas using a SI-aware calculator, such as GNU units [32].

3.1.1 Equations of motion

The relativistic equation of motion of a charged particle in an external electromagnetic field is (see for example [30])

$$\frac{dp^\mu}{d\tau} = \frac{q}{c} F^{\mu\nu} u_\nu \quad \text{with} \quad F^{\mu\nu} = \partial^\mu A^\nu - \partial^\nu A^\mu = \begin{pmatrix} 0 & -\vec{E}^T \\ \vec{E} & \mathbf{B} \end{pmatrix} \quad \text{and} \quad \mathbf{B} = \begin{pmatrix} 0 & -B_z & B_y \\ B_z & 0 & -B_x \\ -B_y & B_x & 0 \end{pmatrix}, \quad (3.5)$$

where p^μ and u_ν are the four vectors of momentum and velocity, and \vec{E} and \vec{B} are the three vectors of the external electric and magnetic field vectors, respectively.

It is convenient to derive a differential equation for the three-velocity, depending on the three-vectors \vec{E} , \vec{B} and \vec{v} . By inserting the explicit expressions (3.2,3.3) for p^μ and u_ν in (3.5),

$$\frac{d\vec{v}}{dt} = \frac{q}{m\gamma} [\vec{E} + \vec{\beta} \times \vec{B} - \vec{\beta} (\vec{\beta} \cdot \vec{E})] \quad \text{with} \quad \vec{\beta} = \frac{\vec{v}}{c}. \quad (3.6)$$

The first term in (3.6) is the electrostatic acceleration, the second term is the Lorentz force of a charged particle inside an external magnetic field, and the third term is preventing any electric field to accelerate a charged particle beyond the speed of light.

3.1.2 Retarded electromagnetic fields

A relativistic description of the dynamics of particles has to take into account the finite speed (of light) with which any change in the dynamical situation of one particle is propagating towards the collision partner. This effect is called retardation and is visualized in figure 3.1.

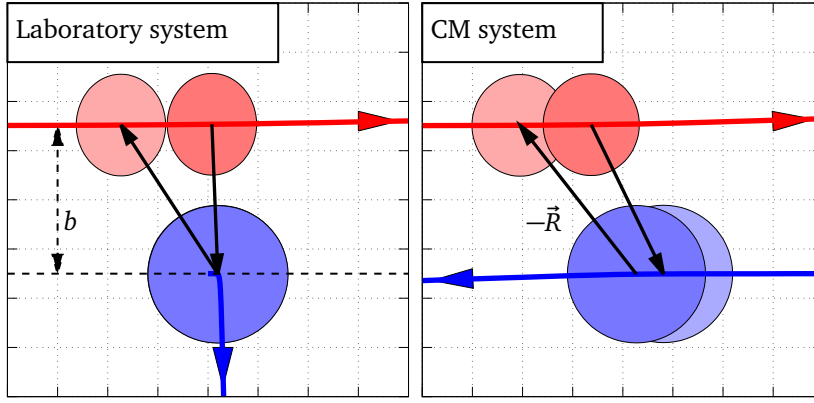


Figure 3.1.: A scattering process of ^{80}Kr (red) at $E_{\text{kin}} = 150 \text{ MeV/u}$ on ^{197}Au (blue) with an impact parameter $b = 15 \text{ fm}$ in different reference frames. Thick colored lines show particle trajectories. Black arrows indicate the retardation effect by pointing from a particle's position at time t (dark circle) to its collision partner's positions at an earlier time t_{ret} .

The derivation for the equations for the retarded electromagnetic fields is given for example in [33]. It starts from the covariant expression of the four vector potential, the Liénard-Wiechert potentials:

$$A^\mu(\vec{r}, t) = \frac{q u_{\text{ret}}^\mu}{u_{\text{ret}}^\nu R_\nu} \quad \text{or} \quad \phi(\vec{r}, t) = \frac{q}{R - \vec{R} \cdot \vec{\beta}_{\text{ret}}} \quad , \quad \vec{A}(\vec{r}, t) = \frac{q \vec{\beta}_{\text{ret}}}{R - \vec{R} \cdot \vec{\beta}_{\text{ret}}} \quad , \quad (3.7)$$

where $\vec{\beta}_{\text{ret}} = \vec{\beta}_{\text{source}}(t_{\text{ret}})$. From (3.7) follow the \vec{E} and \vec{B} fields at the particle's location \vec{r} at time t , caused by the collision partner on a location \vec{r}_{ret} at time t_{ret} :

$$\vec{E}(\vec{r}, t) = \frac{q (\vec{e}_R - \vec{\beta}_{\text{ret}})}{R^2 \gamma^2 (1 - \vec{\beta}_{\text{ret}} \cdot \vec{e}_R)^3} + \frac{q \vec{e}_R \times [(\vec{e}_R - \vec{\beta}_{\text{ret}}) \times \dot{\vec{\beta}}_{\text{ret}}]}{c R (1 - \vec{\beta}_{\text{ret}} \cdot \vec{e}_R)^3} \quad (3.8)$$

$$\vec{B}(\vec{r}, t) = \vec{e}_R \times \vec{E} \quad , \quad (3.9)$$

where $R = |\vec{R}|$, $\vec{e}_R = \frac{\vec{R}}{R}$ and $\vec{R} = \vec{r} - \vec{r}_{\text{ret}}$. The retarded time t_{ret} is implicitly defined by the relation

$$c(t - t_{\text{ret}}) = |\vec{r} - \vec{r}_{\text{ret}}| \quad , \quad (3.10)$$

which expresses the fact that any change in the movement of the collision partner has to propagate some distance with the speed of light c before it can take effect on the motion of another particle at a different location. The position $\vec{r}_{\text{ret}} = \vec{r}_{\text{source}}(t_{\text{ret}})$ is given by the location of the field-creating particle at the retarded time, while the position $\vec{r} = \vec{r}_{\text{probe}}(t)$ refers to the position and time where the field is probed.

Equation (3.8) has two terms, of which the first one describes the field of a non-accelerated charged particle. The second term vanishes for uniform motions, and is proportional to the acceleration. It describes the emission of electromagnetic waves, known as synchrotron radiation. As it turns out in section 3.2.2, this is a minor contribution in the collision problem.

The solution of equation (3.10) is problematic in numerical calculations of particle dynamics, because the simulation will start a point t_0 in time, but information from the past $t_{\text{ret}} < t_0$

are needed to calculate forces at time t_0 . This fact reduces the application of such numerical computations to situations, where at the start of the simulation, any mutual interaction is small enough to be neglected. Fortunately, this is the case for the two-particle collision, where only scattering orbits are of interest. The same technique could not be applied to make relativistic simulations of bound orbits and it is interesting to realize that an apparently simple numerical problem of two moving particles in a bound orbit is so difficult.

3.2 Numerical scattering solutions of the relativistic Kepler problem

The numerical simulation of the relativistic two-body collision process has three main tasks: determination of the retarded time by solving (3.10), implementation of the equations of motion, and the numerical integration of the coupled differential equations.

3.2.1 Implementation

As units of measurement, the program uses zs for time, fm for length, and MeV for energy. In order to solve (3.10) for t_{ret} , the simulated trajectories of both particles have to be memorized from the starting time t_0 until the current simulation time t . Further, it must be possible to get the trajectory at any time $t_{\text{ret}} < t_0$ before the simulation has started by extrapolating the starting conditions into the past, assuming that all interactions that happened were negligible. The simulation is done in discrete steps in time, serving as supporting points of the particle trajectory. Because trajectories need to be computed at any t_{ret} , also between two supporting points, fifth-order splines are used to interpolate between the two closest supporting points to get precise values for r_{ret} and v_{ret} . The order of five is determined by demanding a smooth function of the trajectory itself and its first and second derivative, which correspond to position, velocity and acceleration, respectively. For both particles, the value of t_{ret} has to be determined. This can be done using bisection on a given time interval $[t_{a,i}, t_{b,i}]$ along the trajectory of the collision partner, starting with the full history $[t_{a,0}, t_{b,0}] = [t_0, t]$. Each time the interval is split in two equal parts $[t_{a,i}, t_{m,i}], [t_{m,i}, t_{b,i}]$ with $t_{m,i} = (t_{a,i} + t_{b,i})/2$ and the condition

$$c(t - t_m) < \vec{r}_{\text{probe}}(t) - \vec{r}_{\text{source}}(t_m) \quad (3.11)$$

it is tested. In case (3.11) is true, $t_{b,i+1} = t_{m,i}$, otherwise $t_{a,i+1} = t_{m,i}$. The iteration continues until $t_{b,i} - t_{a,i} < \varepsilon$ and $t_{\text{ret}} = t_{m,i}$ is taken as solution of (3.10). For the following calculations, $\varepsilon = 5 \cdot 10^{-12}$ was used as limit.

Knowing t_{ret} , the coupled equation of motions for the movement of the projectile follow from equation (3.5)

$$\frac{d\vec{r}_{\text{proj}}}{dt} = \vec{v}_{\text{proj}}(t) \quad (3.12)$$

$$\frac{d\vec{v}_{\text{proj}}}{dt} = \frac{q_{\text{proj}}}{m_{\text{proj}} \gamma_{\text{proj}}(t)} [\vec{E}(\vec{r}_{\text{proj}}, t) + \vec{\beta}_{\text{proj}} \times \vec{B}(\vec{r}_{\text{proj}}, t) - \vec{\beta}_{\text{proj}} (\vec{\beta} \cdot \vec{E}(\vec{r}_{\text{proj}}, t))] \quad (3.13)$$

$$\frac{d\tau_{\text{proj}}}{dt} = \frac{t}{\gamma_{\text{proj}}(t)}. \quad (3.14)$$

The equations for the target are obtained by swapping the indices *proj* and *targ* in equations (3.12-3.14). Given that the trajectory is confined to a two-dimensional plane, the vector quantities in (3.12-3.14) are non-zero in two dimensions, and including the equation for the proper time (3.14), there are in total ten coupled differential equations.

Integrating these coupled differential equations is nowadays easy, using advanced and freely available numeric libraries, such as the GNU scientific library (GSL) [34]. The GSL provides a variety of integration algorithms for coupled differential equations of which the *Explicit embedded Runge-Kutta-Fehlberg (4, 5) method* (RKF45) [35] was chosen. The numbers in the parentheses refer to the order of the integration (fourth order) and error estimation (fifth order), respectively. RKF45 provides adaptive step size control to achieve the desired precision in the end result and is advertised in the GSL manual as good general purpose integrator.

3.2.2 Verification of the simulation

In each numerical simulation, it is important to ensure the correctness and to quantify the accuracy of the results. If analytic solutions for special cases of the problem are known the simulation can be precisely tested against them for these cases. In the case of relativistic two-particle collisions, another way to verify a solution is possible. The simulation can run in two different reference frames and the results from both simulations can be compared after boosting them into the same system. Both verification methods are applied in the following and confirm the correctness of the implementation.

Analytic solutions are available in the non-relativistic limit [5] where $1 \gg v/c$, and for extreme forward scattering which is the case for fast projectiles with $v/c \approx 1$, or for projectiles with a large impact parameter. The latter case is covered by the straight-line approximation [31]. For intermediate velocities, where none of the aforementioned conditions is valid, solutions are available [36] for large mass asymmetry, i.e. $m_{\text{targ}} \gg m_{\text{proj}}$.

Rutherford trajectories, the non-relativistic limit

In the first case, the trajectories are given as the classical Kepler hyperbolas. The following parameterization (3.16), using the parameter w defined in (3.15), is taken from [5]

$$t = \frac{a}{v} (b \sinh w + w) \quad , \quad r = a (b \cosh w + 1) \quad , \quad (3.15)$$

$$x = a (\cosh w + b) \quad , \quad y = a \sqrt{b^2 - 1} \sinh w \quad , \quad z = 0, \quad (3.16)$$

and is valid for the center-of-mass (CM) system, where t is the time, $r = |\vec{r}_{\text{proj}} - \vec{r}_{\text{targ}}|$ is the distance of the collision partners, b the impact parameter and the parameter a is half the distance of closest approach of the two particles. This half-distance of closest approach is defined by the strength of electromagnetic repulsion, the initial relative velocity v and the reduced mass $\mu = (m_{\text{proj}} m_{\text{targ}})/(m_{\text{proj}} + m_{\text{targ}})$ as

$$a = \frac{Z_{\text{proj}} Z_{\text{targ}} \alpha \hbar c}{\mu v^2}. \quad (3.17)$$

The scattering angle in the laboratory frame is related to the impact parameter b by

$$\tan\left(\frac{\vartheta}{2}\right) = \frac{a}{b}. \quad (3.18)$$

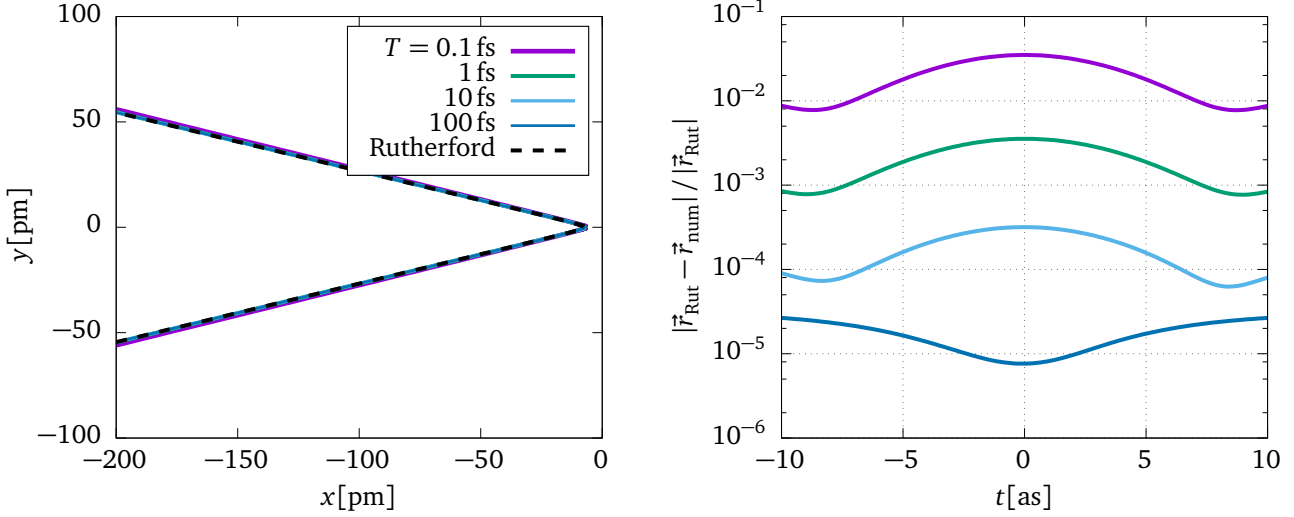


Figure 3.2.: Comparison of the analytic results with numeric solution, based on the relative deviation between the two positions at a given time t . The picture on the left hand side shows an overlay of all simulated trajectories (colored lines) and the corresponding Rutherford trajectory (dashed line). On the right hand side, the deviation of the simulated from the analytic solution is plotted over time. Differently colored lines represent simulations using different integration intervals ($t \in [-T, T]$).

Figures 3.2 and 3.3 show the agreement of the simulation with the Rutherford solution using ^{80}Kr on ^{197}Au at 0.01 MeV/u at an impact parameter of $b = 40\text{ fm}$ with varying integration intervals $t \in [-T, T]$. The required relative precision in the integration routine was 10^{-6} . The simulation was performed in the CM-frame and the comparison was done using the relative distance of projectile and target $\vec{r}_{\text{proj}} - \vec{r}_{\text{targ}}$. In the simulation, the initial projectile velocity is along the x-axis. For a comparison with the analytical result of the Rutherford parameterization, the final simulated trajectory has to be rotated by half the simulated scattering angle $\vartheta/2$. It can be seen, that the agreement is in general good, and that it is important for low velocities to start the integration early enough, i.e with a large value of T , because the momentum change caused by a tiny field at large distances can have a big relative impact if the particle is moving slowly. The agreement is better for smaller velocities, as it is expected by the non-relativistic nature of the analytical solution.

Straight line approximation

The second analytic extreme case is the straight line approximation at high projectile velocity $v/c \approx 1$ or large impact parameter, where the target is assumed to be stationary and the projectile is moving exactly on a straight line with distance b to the target. A scattering angle ϑ can be defined by calculating the transverse momentum transfer Δp due to the field of the target and $\vartheta \approx \Delta p/p$, where $p = \gamma m v$ is the momentum of the incoming projectile. This results in the following formula for the scattering angle (see for example [21])

$$\vartheta_{\text{lab}} = \frac{2 Z_{\text{proj}} Z_{\text{tar}} \alpha \hbar c}{m_{\text{proj}} v^2 \gamma b} \quad (3.19)$$

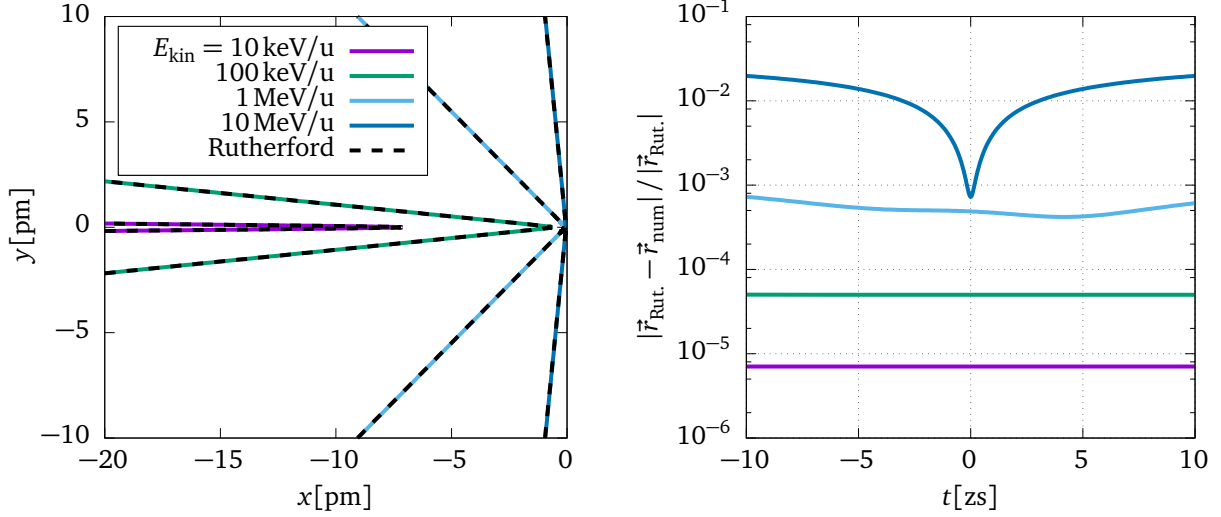


Figure 3.3.: Same situation as in figure 3.2, only the integration interval is constant ($T = 100$ fs), while the kinetic energy of the projectile is varied. Because the simulation is fully relativistic, it agrees with the non-relativistic analytic trajectories only at low energies.

The following discussion is similar to [33]. In the rest frame of the projectile, the resulting field that is caused by the collision partner is the Lorentz-boosted electric field of a stationary charge

$$\vec{E} = q \frac{\vec{r}}{r^3}, \quad \vec{B} = 0, \quad (3.20)$$

where $r^3 = (b^2 + \gamma^2 v^2 t^2)^{3/2}$ for the straight line movement. The transformation of the electric and magnetic field is given by $F'^{\alpha\beta} = \Lambda^\alpha_\mu \Lambda^\beta_\nu F^{\mu\nu}$. Applied on (3.20), this results in the following equations for the \vec{E} and \vec{B} vectors:

$$E_x(t) = \frac{-q \gamma v t}{(b^2 + \gamma^2 v^2 t^2)^{3/2}}, \quad E_y(t) = \frac{q \gamma b}{(b^2 + \gamma^2 v^2 t^2)^{3/2}}, \quad B_z(t) = \frac{q \gamma b \beta}{(b^2 + \gamma^2 v^2 t^2)^{3/2}}, \quad (3.21)$$

while the other components are zero ($E_z = B_x = B_y = 0$). Equations (3.21) are valid in the rest frame of the field-probing particle and the field-creating particle with charge q moves with velocity v along the x-axis at $y = b$. In [31], an improvement to equations (3.21) and (3.19) is introduced by taking into account the displacement of target and projectile, caused by the mutual repulsion in transversal direction during fly-by. The expression for a modified impact parameter, using the reduced mass μ , is given as

$$b' = b + \frac{\pi}{2} \frac{Z_{\text{proj}} Z_{\text{targ}} \alpha \hbar c}{\mu v^2 \gamma}. \quad (3.22)$$

In order to make a quantitative comparison of the simulation results with the straight line approximation, not the points on the trajectory but rather the field and the scattering angle is chosen. The simulated fields have to be transformed into the rest frame of the particle on which they act and have to be evaluated at the proper time τ (3.14) of the particle. In this case, the

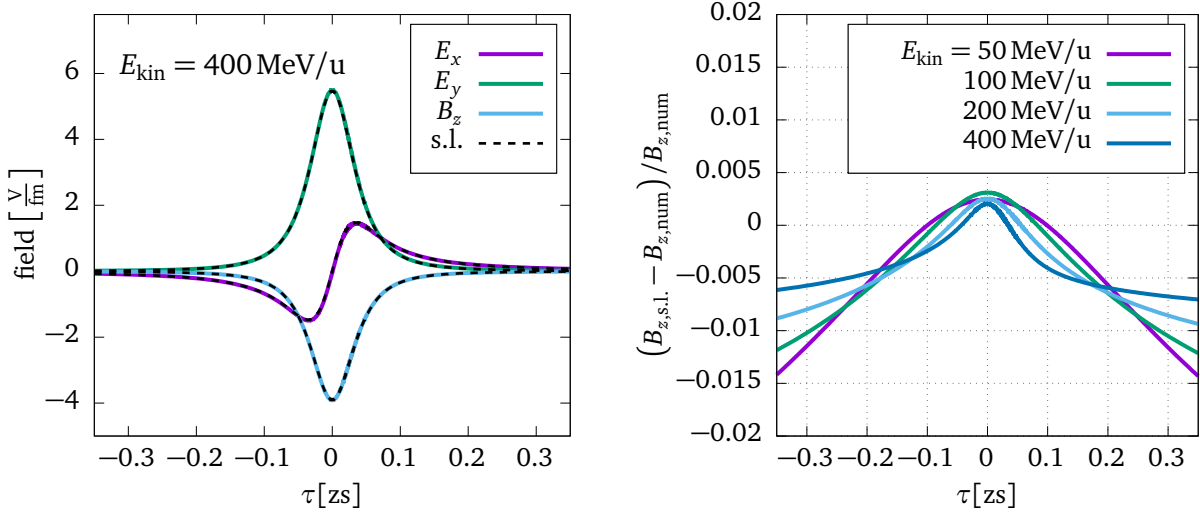


Figure 3.4.: On the left hand side, the strength of the electric and magnetic field components are plotted in the projectile rest frame for an initial kinetic energy of 400 MeV/u. Colored lines represent numerical results of the relativistic simulation, and the dashed, black lines are calculated using equations (3.21) of the straight line (s.l.) approximation with the displaced impact parameter from equation (3.22). The relative difference of simulated and analytic field strengths can be seen on the right hand side.

agreement is good for very large velocities, as can be seen in figure 3.4. The projectile and target isotopes were the same as described in figure 3.2. The impact parameters were chosen to be $b = 15 \text{ fm}$ and the x-axis points in the direction of the initial projectile velocity. Better agreement is achieved for higher impact energies, as the straight-line approximation becomes more accurate. The deviation from the straight-line field (3.21) and simulation is below 1% in the time frame of close proximity. The agreement confirms a correct implementation of the equations of motion and Lorentz transformation of the fields. Note that figure 3.4 looks identical, regardless of being simulated in the laboratory frame or the center-of-mass (CM) frame of reference. In order to get a full overview of the transition between Rutherford and straight line approximation, one can plot the resulting scattering angle over the kinetic energy of the projectile. This is done in figure 3.5 in the laboratory and in the center-of-mass frames. Figure 3.6 shows the same situation at a fixed incident kinetic energy of the projectile and various impact parameter. The simulation agrees with the approximative results at the low and high energy limit. Another valuable information that can be obtained from these pictures is the range of validity of the approximations and the quantitative difference to a rigorous numerical simulation without approximations. The performance of both, Rutherford and straight-line approximations, is surprisingly good. For the considered case, only in the energy range between 50 MeV/u and 300 MeV/u the deviation from the true scattering angle is larger than 1% and stays below 5% for all energies if the correct approximation is chosen.

The infinite mass limit

The relativistic two-particle scattering problem has an analytic solution if one of the two masses is much larger than the other one ($m_{\text{tar}} \gg m_{\text{proj}}$), or equivalently, if one mass is infinite [37]. In this approximation, the equations (3.15) and (3.16) are valid, only the half distance of closest approach a has to be replaced by a/γ , where gamma is $\gamma = 1/\sqrt{1 - \beta^2}$. This is used as a

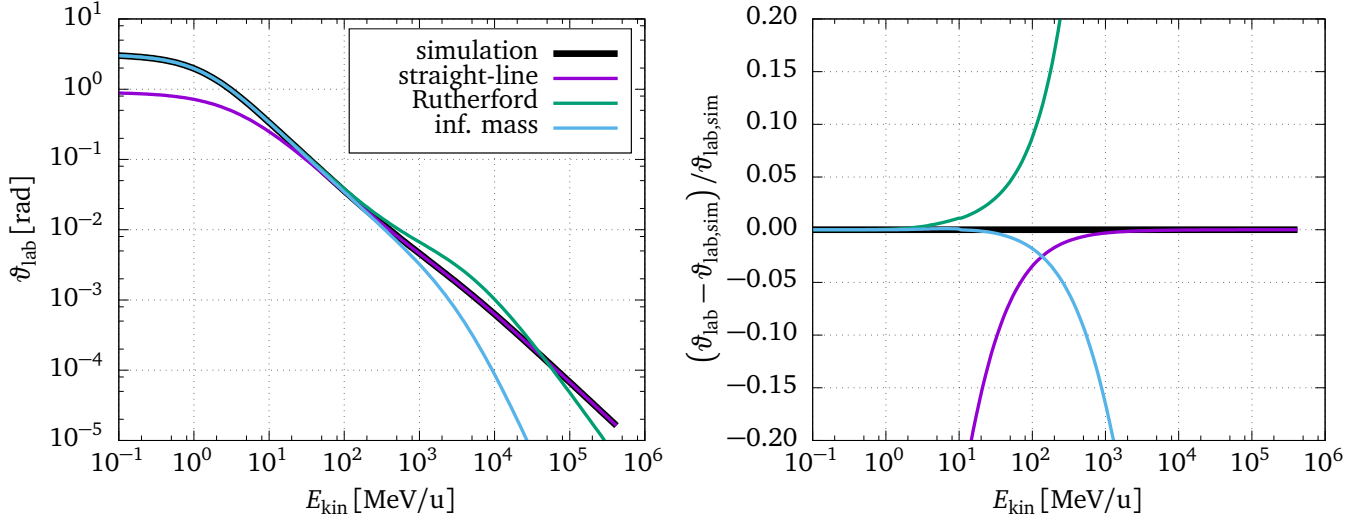


Figure 3.5.: A comparison of the projectile scattering angle and its dependence on the incident kinetic energy of the projectile in the laboratory frame. The picture on the left (right) hand side shows the scattering angle in the laboratory (CM) frame, respectively. The case labeled inf. mass is the analytic solution of the infinite mass limit.

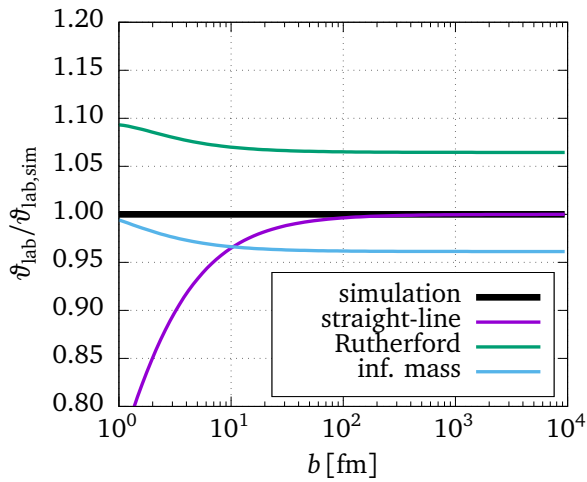


Figure 3.6.: Comparison of the scattering angle ϑ from equation (3.22) with the simulation for $E_{\text{kin}} = 100 \text{ MeV/u}$ and a ^{80}Kr projectile on ^{197}Au target. The straight-line approximation gets closer to the simulation results for bigger impact parameters where the assumption of forward scattering is better fulfilled.

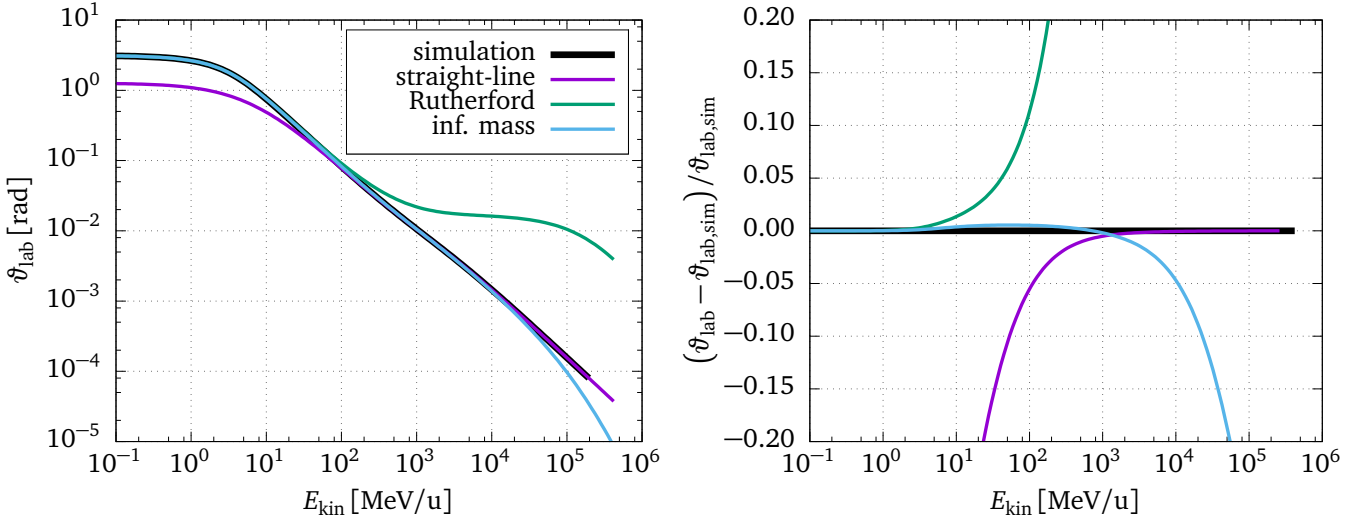


Figure 3.7.: The plot shows the good agreement in the scattering angle ϑ of the infinite mass approximation and the relativistic simulation up to ≈ 10 GeV/u in the case of a proton colliding with ^{208}Pb with $b = 15$ fm at various initial energies.

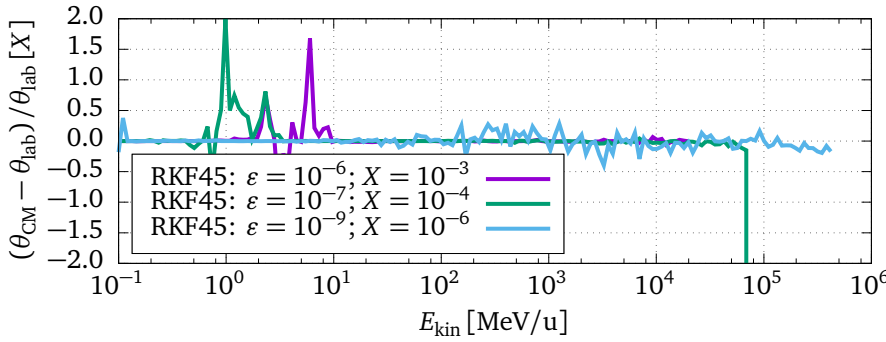


Figure 3.8.: Relative difference of the scattering angles from simulations in laboratory and CM frame. The quantity ε is the relative accuracy requested from the RKF45 solver.

third test case for the numerical solution. One of the most extreme cases is the elastic scattering of a proton on ^{208}Pb , of which the simulation results can be seen in figure 3.7 over a wide range of energies with a fixed impact parameter.

Consistency of the results obtained in different reference systems

The three previous methods rely on the comparison of simulation results for special cases to the analytic results that are available for these cases. An independent method of verification of results does not rely on special cases. The simulation can be executed in two different reference frames and both results can be compared after they have being transformed into the same reference frame. This will be shown here by comparing the obtained scattering angles of a simulation in the laboratory frame and the center of mass frame. Given the momentum four vector (3.3), one can see that $\vec{v} = \vec{p}/(Ec)$. In the case of a two-particle system, the velocity of the center of mass can be determined in the same way from the total momentum four vector $p_{\text{proj}} + p_{\text{targ}}$. One obtains

$$\vec{v}_{\text{CM}} = c \frac{\vec{p}_{\text{proj}} + \vec{p}_{\text{targ}}}{E_{\text{proj}} + E_{\text{targ}}}. \quad (3.23)$$

Figure 3.8 shows a comparison of the scattering angle ϑ_{CM} , obtained in a simulation in the CM frame and in the laboratory frame. For lower energies, an explanation of the deviation might be a too short integration interval. For energies above 10 GeV/u the stability of the integration becomes an issue. Too small solver accuracy does not yield a stable solution in this region. This is, however, not relevant for any practical purpose.

3.3 Coulomb excitation

Coulomb excitation is the process of exciting intrinsic degrees of freedom of a nucleus during the collision with another nucleus by means of the time dependent mutual electromagnetic field that is created by the collision partner. In the semi-classical theoretical description of the Coulomb excitation process the particle movement can be approximately described in terms of classical trajectories instead of quantum mechanical wave functions. The Sommerfeld parameter is a quantitative measure that indicates whether that description is valid. It compares the dimension of the De Broglie wave length $\lambda = h/p = h/(\mu v)$ of the particles with their minimum spatial separation in a collision process. In this case the spatial separation is given by the distance of closest approach in a central collision, and the Sommerfeld parameter is defined in the non-relativistic case as

$$\eta = \frac{2 \alpha \hbar c Z_{\text{proj}} Z_{\text{targ}}}{\hbar v}. \quad (3.24)$$

For $\eta \gg 1$ a description of the excitation process with classical trajectories is justified. This is the case for practically all Coulomb excitation experiments with heavy ions. For example, in the collision of ^{80}Kr on ^{197}Au at $v = 0.5 c$ the Sommerfeld parameter is $\eta \approx 102$. Further, the semi-classical description of Coulomb excitation has two main ingredients: a particle trajectory with a time dependent perturbative field, and the quantum mechanical tool of time dependent perturbation theory. The trajectories and the corresponding fields have been calculated in the previous section and are available in the most general form as a result of the numerical calculation. In the following, the time dependent perturbation theory is discussed.

3.3.1 Time dependent perturbation theory

Quantum mechanical perturbation theory describes transition probabilities between eigenstates of an unperturbed system

$$\hat{H}_0 |n\rangle = E_n |n\rangle, \quad (3.25)$$

induced by a small, time dependent, additional potential $\hat{V}(t)$, see for example chapter 41 of [38]. The resulting transition probability is given by the expression

$$c_{n \rightarrow m} = \frac{1}{i \hbar} \int_{-\infty}^{\infty} e^{i \omega_{mn} t} \langle m | \hat{V}(t) | n \rangle dt \quad \text{with} \quad \omega_{mn} = \frac{E_m - E_n}{\hbar}. \quad (3.26)$$

If more than one excited level is of relevance, one can write a system of differential equations for the time evolution of the occupation probabilities $c_n(t)$, the so called coupled channel equations

$$\frac{dc_m(t)}{dt} = \frac{1}{i \hbar} \sum_n e^{i \omega_{mn} t} \langle m | \hat{V}(t) | n \rangle c_n(t). \quad (3.27)$$

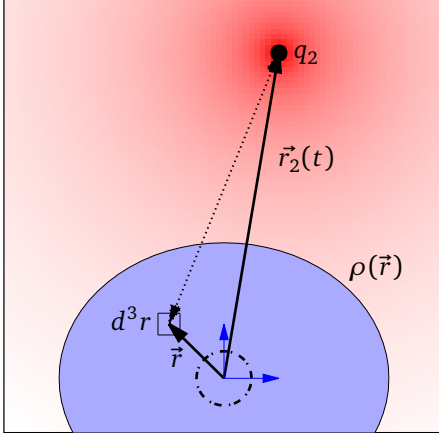


Figure 3.9.: Schematic visualization of the interaction of a nucleus with the scalar monopole field of its collision partner. A small volume element inside the charge distribution $\rho(\vec{r})$ interacts with the point charge q_2 . This has to be integrated over the charge distribution in order to get the total interaction. The strength of the potential is indicated by the red color being proportional to $\ln(\phi)$. The blue area indicates the charge distribution, which has the coordinate system (blue arrows) in its center. The dotted circle represents the numerical integration on a sphere with radius $r_\varepsilon = 0.1 \text{ fm}$ of equation (3.33)

The time dependent potential energy $\hat{V}(t)$ represents the interaction of the nucleus with the electromagnetic field that is created by the collision partner. It has an electrical and a magnetic component of which only the electrical one is considered here:

$$V(t) = V_E(t) + V_M(t) \quad \text{with} \quad V_E(t) = \int \phi(t, \vec{r} - \vec{r}_2) \rho(\vec{r}) d^3r \quad (3.28)$$

$V_E(t)$ depends on the charge distribution ρ of the excited nucleus and the time dependent scalar potential ϕ of the collision partner, which is located at position \vec{r}_2 from the perspective of the excited nucleus. The expressions are all valid in a coordinate system that is at the center of the charge distribution of the excited nucleus. The situation is shown schematically in figure 3.9 for the scalar potential. Note that this is a simplified picture, because retardation is neglected. The detailed way of calculating the potential including retardation is described in section 3.3.2.

Another way of describing the potential energy is in terms of the contributions from different multipole moments $M_{E\lambda\mu}$ of the charge distribution

$$V(t) = \sum_{\lambda\mu} S_{E\lambda\mu}(t) M_{E\lambda\mu}, \quad (3.29)$$

where $S_{E\lambda\mu}(t)$ are the multipole components of the field at the center of the charge distribution. This becomes clear by looking at the examples of electrical monopole and dipole: the potential energy of a charge distribution with only an electrical monopole moment inside an electrical potential only depends on the height of the potential function at the center of the charge distribution, i.e. the monopole moment of the potential function at that point, while the potential energy of a charge distribution with only an electrical dipole moment depends only on the gradient of the potential function, i.e. the dipole moment of the potential function.

In spherical coordinates, the classical multipole moments of an electric charge distribution are defined as

$$M_{E\lambda\mu} = \int r^\lambda Y_{\lambda\mu}^*(\vartheta, \varphi) \rho(\vec{r}) d^3r. \quad (3.30)$$

In quantum mechanics, the multipole moments are expectation values of the multipole operator

$$\begin{aligned} M_{E\lambda\mu} &= \langle m | \hat{M}_{E\lambda\mu} | n \rangle \\ &= \langle I_m, M_m | \hat{M}_{E\lambda\mu} | I_n, M_n \rangle = (-1)^{I_n - M_n} \begin{pmatrix} I_n & \lambda & I_m \\ -M_n & \mu & M_m \end{pmatrix} i^{(-\lambda)} \underbrace{\langle I_m || i^\lambda \hat{M}_{E\lambda} || I_n \rangle}_{\in \mathbb{R}}, \end{aligned} \quad (3.31)$$

which can be expressed in terms of the reduced matrix elements using the Wigner-Eckard theorem. The factor i^λ in the reduced matrix elements is a phase convention in the nuclear wave function and makes the matrix elements real numbers, as described in [5, appendix E]. These matrix elements fulfill the following symmetry relation

$$\langle I_m || i^\lambda \hat{M}_{E\lambda} || I_n \rangle = (-1)^{\lambda+I_n-I_m} \langle I_n || i^\lambda \hat{M}_{E\lambda} || I_m \rangle. \quad (3.32)$$

The multipole components of an electrical potential can be calculated by

$$S_{E\lambda\mu}(t) = \lim_{r \rightarrow 0} \frac{1}{r^\lambda} \int_{\Omega} \phi(t, \vec{r}(\vartheta, \varphi, r)) Y_{\lambda\mu}(\vartheta, \varphi) d\Omega. \quad (3.33)$$

In the case of Coulomb excitation, the time dependence of the potential function is created by the relative movement of the nuclei along their orbits. Therefore, the quantity $S_{E\lambda\mu}(t)$ is also called *orbital integral*. There are also contributions from the magnetic multipoles $M_{M\lambda\mu}$ and magnetic orbital integrals $S_{M\lambda\mu}(t)$, which are not discussed here.

With (3.32) and (3.29), the coupled channel differential equation (3.27) becomes

$$\frac{dc_m(\tau)}{d\tau} = \frac{\alpha \hbar c}{i \hbar} \sum_{n\lambda\mu} e^{i\omega_{mn}\tau} (-1)^{I_n-M_n} \begin{pmatrix} I_n & \lambda & I_m \\ -M_n & \mu & M_m \end{pmatrix} i^{(-\lambda)} \langle I_m || i^\lambda \hat{M}_{E\lambda} || I_n \rangle S_{E\lambda\mu}(\tau) c_n(\tau). \quad (3.34)$$

Note that the equations (3.34) are correct even in the relativistic case because they are now written in terms of the proper time τ in the rest frame of the excited nucleus, and that the potential ϕ has to be transformed into the rest frame as well before calculating $S_{E\lambda\mu}$.

One general and helpful observation about coupled channel calculations can be found if the system of equations (3.27) is written in terms of a matrix multiplication:

$$\frac{d\vec{c}(t)}{dt} = M(t)\vec{c}(t) \quad \text{with} \quad M_{nm}(t) = \frac{1}{i\hbar} e^{i\omega_{mn}t} \langle m | \hat{V}(t) | n \rangle. \quad (3.35)$$

The matrix M has to fulfill one property that can be derived by demanding that the sum of occupation probabilities $\sum_m |c_m|^2$ has to be one, as it is the case for any probability distribution. This condition can be written in the vector formulation: $\vec{c}^\dagger \vec{c} = 1$. It follows from this requirement, that the time derivative $\frac{d}{dt} \vec{c}^\dagger \vec{c}$ must vanish, which in turn requires $M_{ij} = -M_{ji}^*$, i.e. M has to be anti-hermitian.

3.3.2 Numerical solution

In order to solve (3.34) in the most general case, the same GSL integration routine that was already successfully applied to the trajectories is used again. However, one can not rely on the adaptive step size control of the integrator, because the interaction strength is nonzero only for a very short period of time around the point of closest approach. A free running integrator would increase the step size, because it is integrating zero all the time. Then it would jump with large steps over the relevant part of the time evolution. In order to prevent this, the time steps of the trajectory calculation are imposed to the algorithm. Only if the algorithm demands a smaller

step size, which can happen around the point of closest approach, the step size request of the algorithms is granted. The integration routine works with the simulation frame time variable t , so the set of differential equations has to be transformed from the time τ in the rest frame of the excited nucleus to the simulation time t by

$$\frac{dc_m}{dt} = \frac{dc_m}{d\tau} \frac{d\tau}{dt} = \frac{dc_m}{d\tau} \frac{1}{\gamma}, \quad (3.36)$$

where γ is the relativistic factor of the excited nucleus in the simulation frame of reference. The potential ϕ can be calculated from the relative position and velocity that is available in the simulation frame of reference at any time t . If at the beginning of the simulation at t_0 only an initial state n (e.g. the ground state) was populated $c_m = \delta_{mn}$, then, at the end of the simulation (t_{end}), the occupation probabilities represent what is also called *excitation amplitude*

$$a_{n \rightarrow m} = c_m(t_{\text{end}}). \quad (3.37)$$

The remaining problem is the correct numerical calculation of the orbital integral $S_{E\lambda\mu}(\tau)$ in (3.33). The limit $r \rightarrow 0$ can not be realized. Instead, a small value of $r_\epsilon = 0.1 \text{ fm}$ is used. The spherical surface integral (3.33) is calculated using the Lebedev-Laikov quadrature [39], where the integral is approximated by a finite sum over points $\vec{r}_{\text{LL}}(\vartheta_i, \varphi_i)$ on the surface of a unit sphere and corresponding weights:

$$\int_{\Omega} f(\vec{r}(\vartheta, \varphi)) d\Omega \approx 4\pi \sum_i w_i f(\vec{r}_{\text{LL}}(\vartheta_i, \varphi_i)). \quad (3.38)$$

A Fortran program [40] was used to calculate tables of points and weights. For each Lebedev-Laikov integration point $\vec{r}_{\text{proj}}(t) + \vec{r}_{\text{LL}}(\vartheta_i, \varphi_i)$ on the sphere at simulation time t , the potential ϕ from equation (3.7) has to be calculated. The retarded time $t_{\text{ret}} = t_{\text{ret}}(t, \vec{r}_{\text{probe}}(t) + r_\epsilon \vec{r}_{\text{LL}})$ at which the velocity and position of the collision partner has to be evaluated is different for different points on the sphere. It depends on the simulation time t , as well as the position $\vec{r}_{\text{proj}} + \vec{r}_{\text{LL}}$ where the field is probed, i.e. the position of the field probing nucleus plus the current Lebedev-Laikov integration point on the integration sphere. When the retarded time for the integration point is known, the two relevant quantities \vec{R} and β_{ret} for the evaluation of (3.7) can be determined. After the lookup of these values in the history of the trajectory, they have to be transformed into the rest frame of the excited nucleus. This is done by forming the following four vectors, as in (3.1) and (3.2).

$$(u_{\text{ret}}')^\mu = \gamma c \begin{pmatrix} 1 \\ \vec{\beta}_{\text{source}}(t_{\text{ret}}) \end{pmatrix} \quad \text{and} \quad (R_{\text{ret}}')^\mu = \begin{pmatrix} c(t - t_{\text{ret}}) \\ \vec{r}_{\text{probe}}(t) + r_\epsilon \vec{r}_{\text{LL}} - \vec{r}_{\text{source}}(t_{\text{ret}}) \end{pmatrix}. \quad (3.39)$$

Note that R'^μ is a light-like Lorentz vector. A multiplication of both four vectors with the Lorentz boost matrix $\Lambda^\mu{}_\nu(\beta_{\text{probe}})$ transforms them into the rest frame of the excited, field probing nucleus

$$R^\mu = \Lambda^\mu{}_\nu R'^\nu \quad \text{and} \quad u^\mu = \Lambda^\mu{}_\nu u'^\nu, \quad (3.40)$$

and the quantities \vec{R} and $\vec{\beta}_{\text{ret}}$ can be extracted from the boosted vectors. Here, R^μ is a “real” four vector (with four components) because the integration sphere has points outside the reaction plane.

For completeness, the general Lorentz boost matrix is given here as a function of the three velocity $\vec{\beta} = \vec{v}/c$. Only the matrix elements on and on top of the diagonal are given, because the matrix is symmetric.

$$(\Lambda^\mu{}_\nu) = \begin{pmatrix} \gamma & -\gamma\beta_1 & -\gamma\beta_2 & -\gamma\beta_3 \\ \dots & 1 + (\gamma-1)\frac{\beta_1^2}{|\vec{\beta}|^2} & (\gamma-1)\frac{\beta_1\beta_2}{|\vec{\beta}|^2} & (\gamma-1)\frac{\beta_1\beta_3}{|\vec{\beta}|^2} \\ \dots & \dots & 1 + (\gamma-1)\frac{\beta_2^2}{|\vec{\beta}|^2} & (\gamma-1)\frac{\beta_2\beta_3}{|\vec{\beta}|^2} \\ \dots & \dots & \dots & 1 + (\gamma-1)\frac{\beta_3^2}{|\vec{\beta}|^2} \end{pmatrix} \quad (3.41)$$

3.3.3 Analytic solution for the straight line approximation

For fast projectiles, an analytical solution of (3.26) can be obtained by using the electromagnetic potential from equation (3.21), assuming only single step excitation, as shown in [31]. The result is an analytical solution for the excitation amplitudes

$$a_{n \rightarrow m} = -i \frac{Z \alpha \hbar c}{\hbar v \gamma} \sum_{\pi \lambda \mu} G_{\pi \lambda \mu} \left(\frac{c}{v} \right) (-1)^\mu K_\mu(\xi(b')) \sqrt{2\lambda+1} k^\lambda \langle I_m, M_m | \hat{M}_{\pi \lambda \mu} | I_n, M_n \rangle, \quad (3.42)$$

where for electric excitation ($\pi = E$) and $\mu \geq 0$

$$G_{E\lambda\mu} \left(\frac{c}{v} \right) = i^{\lambda+\mu} \frac{\sqrt{16\pi}}{\lambda(2\lambda+1)!!} \left(\frac{(\lambda-\mu)!}{(\lambda+\mu)!} \right)^{1/2} \left(\left(\frac{c}{v} \right)^2 - 1 \right)^{-1/2} \\ \times \left(\frac{(\lambda+1)(\lambda+\mu)}{2\lambda+1} P_{\lambda-1}^\mu \left(\frac{c}{v} \right) - \frac{\lambda(\lambda-\mu+1)}{2\lambda+1} P_{\lambda+1}^\mu \left(\frac{c}{v} \right) \right) \quad (3.43)$$

and for magnetic excitations ($\pi = M$) and $\mu \geq 0$

$$G_{M\lambda\mu} \left(\frac{c}{v} \right) = i^{\lambda+\mu+1} \frac{\sqrt{16\pi}}{\lambda(2\lambda+1)!!} \left(\frac{(\lambda-\mu)!}{(\lambda+\mu)!} \right)^{1/2} \left(\left(\frac{c}{v} \right)^2 \right)^{-1/2} \mu P_\lambda^\mu \left(\frac{c}{v} \right) \quad (3.44)$$

and for $\mu < 0$

$$G_{E\lambda-\mu} \left(\frac{c}{v} \right) = (-1)^\mu G_{E\lambda\mu} \left(\frac{c}{v} \right) \quad \text{and} \quad G_{M\lambda-\mu} \left(\frac{c}{v} \right) = -(-1)^\mu G_{M\lambda\mu} \left(\frac{c}{v} \right). \quad (3.45)$$

The functions K_μ are the irregular modified cylindrical Bessel function of order μ , and $P_\lambda^\mu(x)$ are the associated Legendre functions evaluated at $x > 1$. The adiabaticity parameter $\xi(b')$ depends on the impact parameter and the incident kinetic energy, and compares the oscillation frequency of the nuclear excitation with the time scale of the electromagnetic perturbation

$$\xi(b') = \frac{\omega b'}{v \gamma}. \quad (3.46)$$

Note that the modified impact parameter b' from equation (3.22) is used to take into account the displacement of the closest approach due to mutual repulsion.

Equation (3.42) will be used here to verify the correctness of program for the numerical integration of the coupled channel equations. Figure 3.10 shows the result of calculating $|a_{n \rightarrow m}|^2$ for the excitation of the 2_1^+ state in ^{80}Kr projectile a ^{197}Au target nucleus at 616.6 keV, a reduced matrix element of $61.8 e\text{fm}^2$, and an impact parameter of $b = 20 \text{ fm}$. The z-axis is pointing into the initial beam direction for both calculations. Numerical results are available for $E_{\text{kin}} < 5 \text{ GeV/u}$. The agreement is good up to the highest possible energy for the $|2, \pm 2\rangle$ sub-states, where a major fraction of the occupation probability is found. Deviations are seen in the small values for the less populated $|2, \pm 1\rangle$ and $|2, 0\rangle$ sub-states. Good agreement can be seen in the values up to $\approx 200 \text{ MeV/u}$, in the sense that the numeric solution approaches the analytic one, if the absolute value is larger than 10^{-4} . The reason for the discrepancy at higher energies for the less populated states is not yet fully understood. The results are insensitive to variations in the finite size of r_e in the tested range of 0.1 fm to 1 fm . Also the number of integration points on the sphere, 110 or 50, does not influence the result in any significant way. The deviation probably points to a small systematic error in the calculation that seems to be relevant only for high energies and small excitation amplitudes.

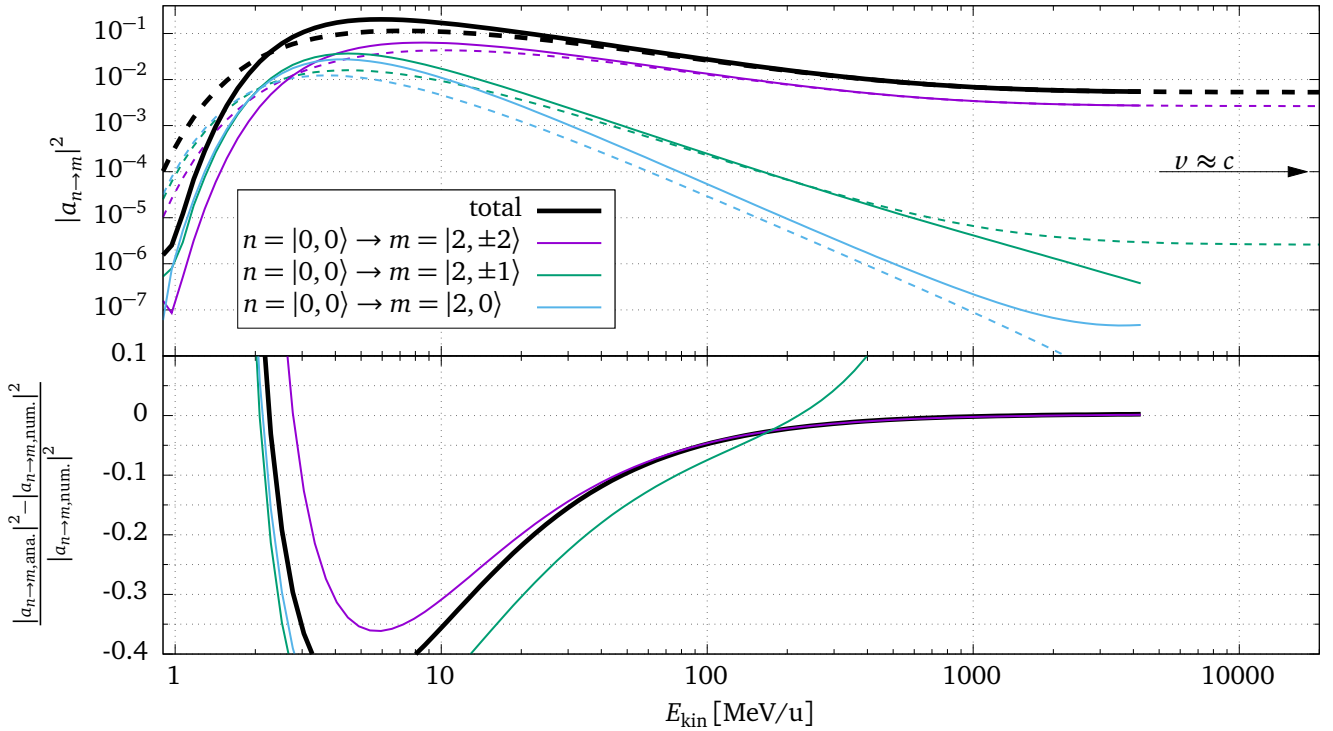


Figure 3.10.: Top panel: Comparison of the numerical results (solid lines) of the excitation amplitudes with equation (3.42) (dashed lines). Bottom panel: Relative differences between numerical and analytical result. The black lines show the sum of all m -sub-states.

3.3.4 Multi-step Coulomb excitation

For intermediate and high energy Coulomb excitation experiments, the fraction of nuclei that show multi-step excitation is negligible. This is shown in figure 3.11, where the sum of the excitation amplitudes of the first 4_1^+ and 2_1^+ states are compared for the same setting as before: ^{80}Kr

projectile, ^{197}Au target, and an impact parameter of $b = 20 \text{ fm}$. The reduced matrix elements are

$$\langle 2_1^+ || i^\lambda \hat{M}_{E\lambda} || 0_1^+ \rangle = 61.8 \text{ efm}^2 \quad \text{and} \quad \langle 4_1^+ || i^\lambda \hat{M}_{E\lambda} || 2_1^+ \rangle = 113.6 \text{ efm}^2, \quad (3.47)$$

obtained from known $B(E2)$ values. It can be seen that the probability of finding the nucleus

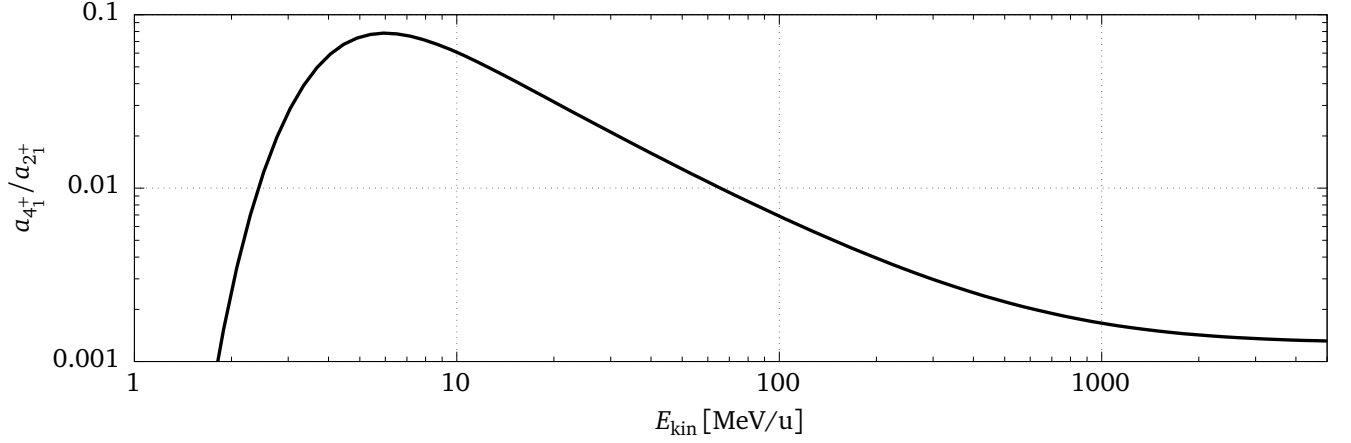


Figure 3.11.: Ratio of 4_1^+ excitation and 2_1^+ excitation in Coulomb excitation of a ^{80}Kr projectile on a ^{197}Au target at various energies. For intermediate and high energy ranges above $\approx 50 \text{ MeV/u}$, the 4_1^+ excitation amplitude is below 1%.

in the 4_1^+ state after the collision is less than 1% for energies above 50 MeV/u and is therefore usually negligible in the analysis of experimental data, because typical experimental errors are larger.

3.3.5 Cross sections

Excitation amplitudes allow to calculate excitation cross sections by integrating them over all possible impact parameters

$$\sigma_{I_n \rightarrow I_m}(b_{\min}) = \frac{2\pi}{2I_n + 1} \int_{b_{\min}}^{\infty} \sum_{M_n, M_m} |a_{|I_n, M_n\rangle \rightarrow |I_m, M_m\rangle}(b)|^2 b \, db, \quad (3.48)$$

where b_{\min} is the minimum possible impact parameter for safe Coulomb excitation. For $b < b_{\min}$, nuclear interaction sets in and starts to interfere with the electromagnetic excitation process. If the initial kinetic energy is below the Coulomb barrier, the nuclei cannot come close enough for nuclear interactions to influence the process, even for $b = 0$. For the projectile energy range between $E_{\text{kin}} = 10 \text{ MeV/u}$ and $E_{\text{kin}} = 3 \text{ GeV/u}$, the cross section for the Coulomb excitation of the first 2_1^+ state in ^{80}Kr on a gold target is shown in figure 3.12. Only the matrix element of the first 2_1^+ state in (3.47) was considered, i.e. no multistep excitation can take place. The minimum impact parameter in the integration of (3.48) was chosen to be $b_{\min} = 15 \text{ fm}$. One can see that for a projectile energy of $E_{\text{kin}} = 150 \text{ MeV/u}$, the deviation of the analytical formula is only in the order of $\approx 3\%$, confirming again that the use of this approximation is justified in this energy region. For a projectile energy of 50 MeV/u the deviation is $\approx 10\%$, which is already above typical experimental uncertainties.

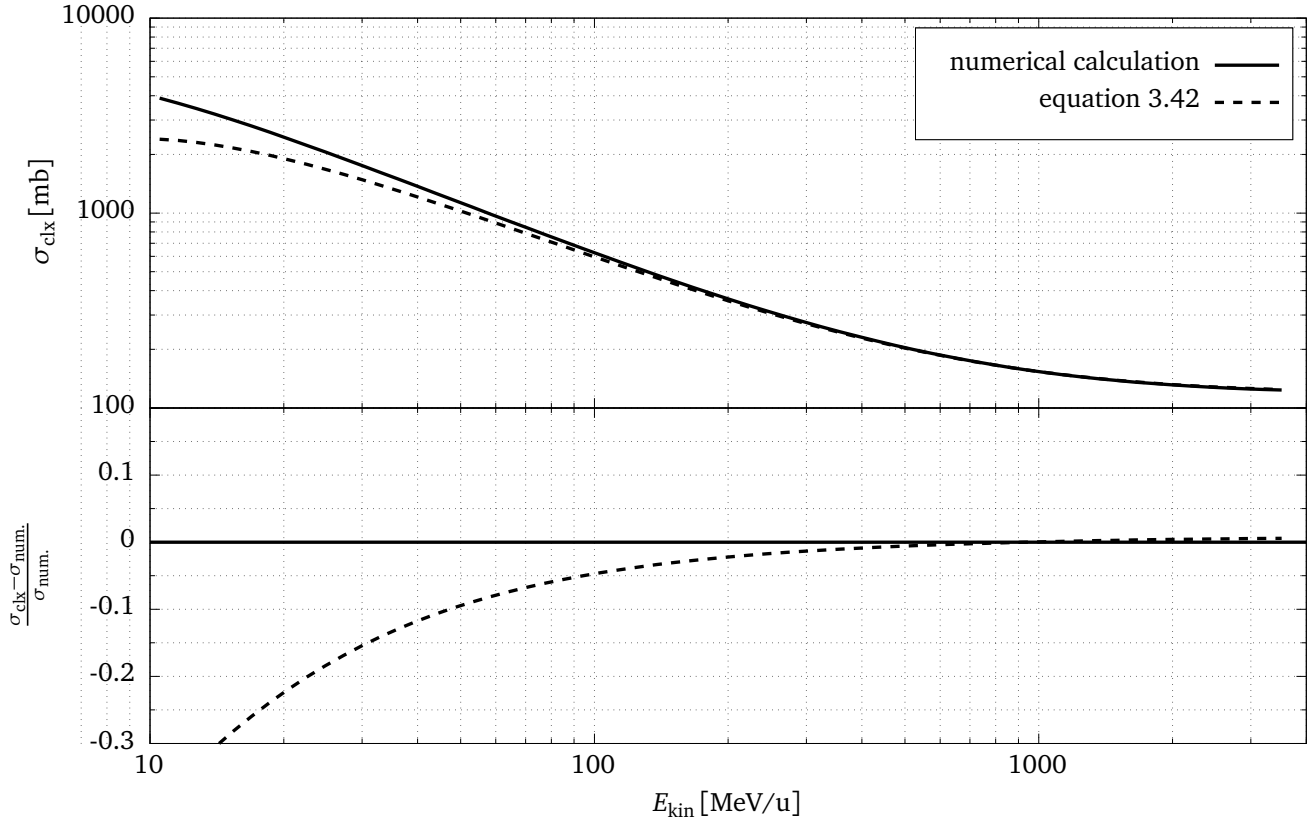


Figure 3.12.: Comparison of cross section calculated numerically (solid line) and using equation 3.42 (dashed line). The top panel shows the absolute values, while the bottom panel shows the relative difference between both calculations. The cross section is underestimated by the analytical formula by more than 5% for projectile energies below 100 MeV/u.

3.3.6 DWEIKO

DWEIKO [41, 42] is a computer program that solves the coupled channel equation numerically using the fields from the straight line approximation (3.21). In addition to the electromagnetic interaction, DWEIKO introduces an optical potential that allows the approximative treatment of nuclear excitations, allows to calculate nuclear-Coulomb interference, and takes into account the exit channels of the reactions when the incoming isotope is destroyed in the course of the nuclear reaction. This optical potential is chosen to be of the Woods-Saxon type. Width and smoothness parameters of this potential can be tuned to match the experimental data, as can be seen in figure 5.15 of section 5.6.5. Mainly because of the optical potential, all cross-section calculations for the experiment analysis in the following chapters were done using DWEIKO.

3.3.7 Summary

A computer program was written that calculates a numerical solution to the coupled differential equations for relativistic motion of two colliding, charged, point-like particles in their mutual, retarded electromagnetic fields. The calculation of the trajectories is basically a reimplementation of a similar program that was written as early in 1987 by Matzdorf et al. [30]. On top of

the trajectories, the nuclear coupled channel equation is solved based on the relativistic electromagnetic fields. This has been done for the first time and extends the set of Coulomb excitation programs by one that can calculate the Coulomb excitation process at arbitrary energies. The program has the same limitations as similar programs using a semi-classical description. It is only applicable in cases where the Sommerfeld η parameter is large and the assumption of classical trajectories is justified. In the context of the rest of this work it was used to verify the correctness of the straight line approximation as used in DWEIKO, which was applied to the analysis of experimental data. It might be useful for future precision Coulomb experiments in the energy range between 10 MeV/u and 50 MeV/u, where the gap between Rutherford trajectories and straight line approximation is. It was written in the modern D programming language in a modular way that makes it easy to extend its functionality. The program is available under the GPL license on github [43].

The program as it is now, cannot calculate magnetic excitations. They are planned to be added in the future. Finally, it would be good to investigate the reason for the discrepancy between the numerical results of the program and the analytic straight line solution for the low excitation amplitudes above ≈ 200 MeV as shown in figure 3.10.

4 The PreSPEC experimental setup

The PreSPEC experimental setup [44] is the successor of the RISING (Rare ISotopes INvestigation at GSI) fast beam setup [21], and the predecessor of the HISPEC (HIGH resolution in flight SPECtroscopy) [45] setup at the presently constructed FAIR (Facility for Antiproton and Ion Research) [46] facility. All these experiments are designed to perform in-beam γ -ray spectroscopy of exotic nuclei after in-flight separation with the GSI fragment separator (FRS) [47] for RISING and PreSPEC, and with the Super-FRS [48] for HISPEC. The particle beam is injected into the FRS from the SIS18 synchrotron with kinetic energies up to 1 GeV/u for Uranium [49]. Typical production targets at the beginning of the FRS have a thickness of a few mm and are made of beryllium. A drawing of the complete setup at GSI, including the FRS beam line, is shown in figure 4.1.

The primary beam and the reaction products from the production target enter the ion optics of the FRS where a subset of reaction products is separated from the primary beam, and the selected particles are identified one by one. The need to identify each individual beam particle limits the rate of usable secondary beams of in-flight separators to about 10^5 particles per second (pps), depending on the acceptable rates in the particle identification detectors. At the end of the FRS, the beam passes a secondary target, where Coulomb excitation or fragmentation reactions happen between the beam and target nucleus. In order to compensate the low beam intensity, this target has to be very thick, typically in the order of few hundred mg/cm². Such thick targets are possible because of the high particle velocity in the order of $v \approx 0.5c$. A High-Purity Germanium detector array is measuring the emitted γ -radiation in coincidence with the identified beam particles. In order to select different reaction channels in the target, the outgoing particles have to be identified. This is done with the Lund-York-Cologne CALorimeter (LYCCA), which combines a time-of-flight (ToF) measurement and a ΔE - E telescope to perform mass and charge identification. The achievable mass resolution in LYCCA allows unambiguous isotope identification only up to mass $A \approx 50$. In HISPEC, it is planned to use LYCCA in combination with a magnetic dipole spectrometer, in order to enhance the mass resolution. LYCCA was commissioned in October 2010, week 37, during the S363 beam time at GSI. Since then, a number of experiments has been performed using LYCCA (see table 4.1). A significant part of this thesis was devoted to the improvement and simplification of the calibration and data analysis of this device. This is discussed in detail in the following section 4.1.

For γ -ray detection, Euroball cluster detectors were used as HPGe detector array in the PreSPEC experiments in the years 2010 and 2011. From 2012 on, the AGATA (Advanced GAMMA-ray Tracking Array) [22] was mounted. AGATA is an array of 36-fold segmented HPGe (high-purity germanium) detectors that are capable of measuring the individual interaction points \vec{x}_γ of the detected γ -quanta inside the crystal material with a spatial accuracy of $\Delta\vec{x}_\gamma \leq 5$ mm, using pulse-shape analysis techniques. AGATA will be described in section 4.2.

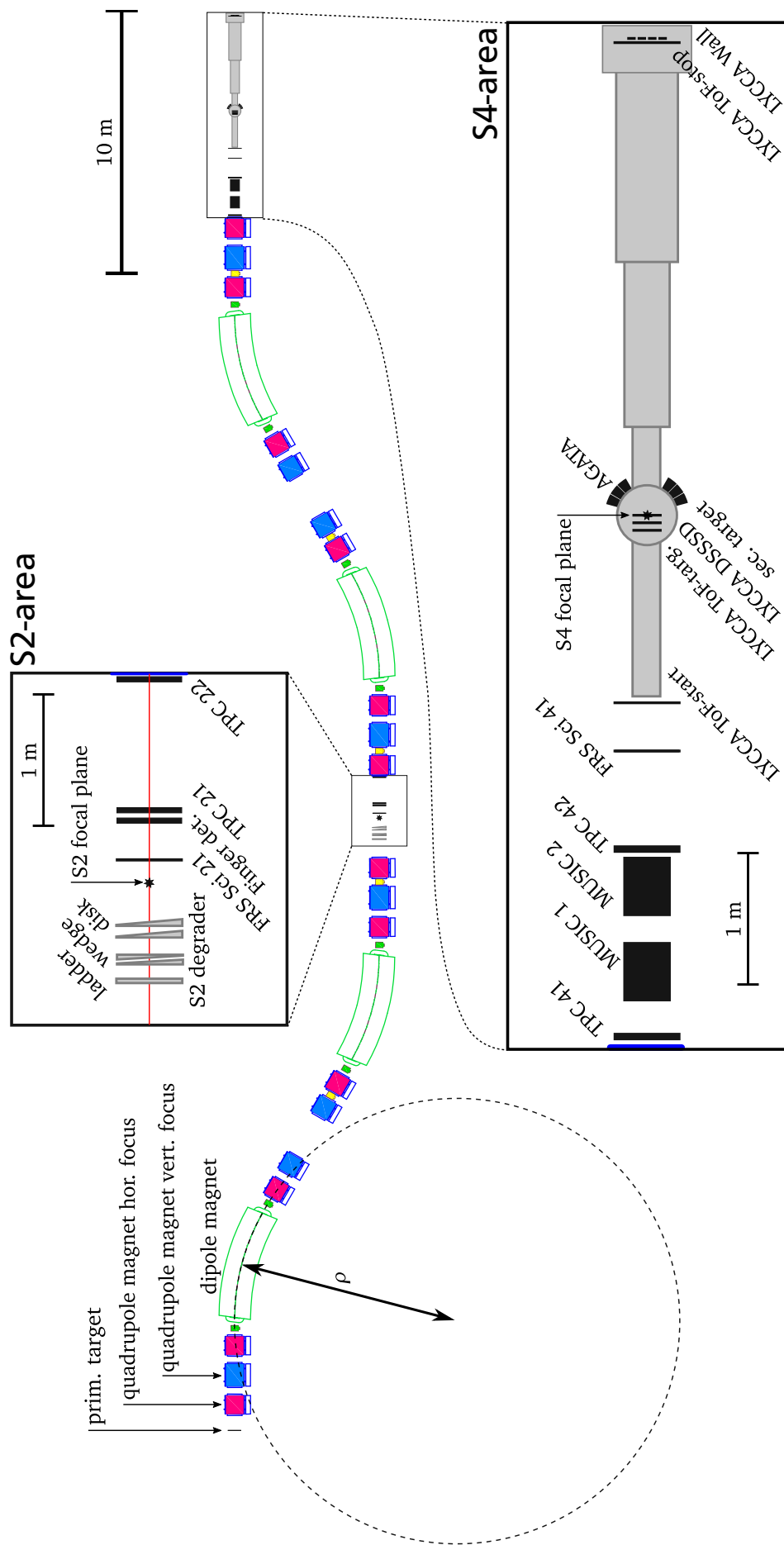


Figure 4.1.: A schematic drawing of the experimental setup, including all FRS magnets (dipoles green, quadrupoles blue and magenta), and all active detector elements (black). A circle with 8.5 m radius is drawn to indicate the curvature of the beam inside the dipole. The detectors are schematically represented by the outline of their housing. S2 degrader setup is just schematically drawn. The ordering of the individual elements and the exact positions are not important. Absolute distances are accurately drawn for all active detectors. The SIS-18 beam will enter from the left-hand side. S2 and S4 areas are enlarged by a factor of 5 inside the black boxes. The arrangement of HP-Ge AGATA detectors is only schematic, while the size of the individual crystals is accurate. This figure is based on a technical drawing of the FRS [50]

Experiment	Date	prim. Beam	Description
S363 (Eu-Cl)	Sep. 2010 Oct. 2010	^{64}Ni ^{86}Kr	LYCCA-0 commissioning
S369 (Eu-Cl)	Oct. 2010	^{238}U	^{88}Kr Coulex
S372 (Eu-Cl)	Nov. 2010	^{124}Xe	^{104}Sn Coulex
S378 (Eu-Cl)	May 2011	^{54}Cr	PreSPEC liquid H_2 -target test
S391 (Eu-Cl)	May 2011	^{54}Cr	PreSPEC Plunger/AGATA test
S377 (Eu-Cl)	May 2011	^{36}Ar	Coulex ^{33}Ar
S424 (AGATA)	Apr. 2012 May. 2012 June. 2012	^{50}Ti ^{132}Xe ^{238}U	PreSPEC-AGATA technical commission
S424 (AGATA)	Aug./Sep. 2012	^{80}Kr	PreSPEC-AGATA performance commissioning: Coulex & sec. fragmentation of ^{80}Kr
S429 (AGATA)	Sep./Oct. 2012	^{208}Pb	Coulex of 2_1^+ state in the Pb region
S426 (AGATA)	Oct. 2012	^{86}Kr	Spin-flip transition in ^{85}Br with M1-Coulex
S430 (AGATA)	Oct. 2012	^{86}Kr	Pygmy Dipole resonance in ^{64}Fe
S433 (AGATA)	Oct. 2012	^{58}Ni	Coulex of the band-terminating 12^+ yrast trap in ^{52}Fe
S428 (AGATA)	Nov. 2012	^{238}U	Shape evolution in neutron-rich Zr isotopes $A \approx 96$ Zr with DSAM lifetime measurements
S431 (AGATA)	Nov. 2012	^{238}U	Proton hole states in ^{132}Sn and $N = 82$ shell structure
S429 (AGATA)	Mar. 2014	^{238}U	continuation
S426 (AGATA)	Mar. 2014	^{86}Kr	continuation
S430 (AGATA)	Mar. 2014	^{86}Kr	continuation
S434 (AGATA)	Apr. 2014	^{58}Ni	$B(E2)$ measurements in the $A = 46$ isobaric $T = 1$ triplet

Table 4.1.: Overview of PreSPEC, PreSPEC-AGATA, and related experiments at GSI. The data shown in this work is from the PreSPEC-AGATA performance commissioning experiment (bold font). From 2012 on, an increasing number of AGATA detectors were used as HPGe γ -ray spectrometer. Previously, Euroball Cluster detectors were used.

4.1 The Lund-York-Cologne Calorimeter Array (LYCCA)

The task of the LYCCA detector system is the identification and tracking of outgoing projectiles after the reaction in the secondary target. It is one of the detector systems for the HISPEC experiments, which are part of the NUSTAR collaboration at GSI/FAIR. LYCCA was build as the successor of the Calorimeter Telescope (CATE) [51] of the RISING [21] in-beam γ -ray spectroscopy experiments at GSI. CATE was capable of determining the nuclear charge Z and the direction of the outgoing particles using a position sensitive ΔE - E detector array with 9 modules. LYCCA extends this capability with a time-of-flight (ToF) measurement and allows to determine additionally the mass A and the velocity β of the outgoing projectile.

In the previous experiments, good isotope identification was only achieved for experiments with relatively light ions ($A \approx 36$ [52], $A \approx 54$ [23]) and/or for only a fraction of the available detectors [23]. In [53] ($A \approx 96$), no distinct mass peaks could be resolved and in [54] the mass information of LYCCA was not used at all. The difficulty of analyzing LYCCA data arises mainly from the high segmentation of all detector subsystems: 144 CsI-crystals, 16 DSSSD detectors, each with 2×16 strips, and 76 scintillators for the time-of-flight measurement. High segmentation is needed, because smaller detectors are cheaper and easier to build than large ones. Another advantage is the capability to handle higher rates, because the rate is distributed among the individual segments of the CsI and DSSSD arrays. Calibrating a detector system gets more difficult with increased detector segmentation. Therefore, the development of new algorithms to perform a high quality calibration of the different LYCCA detector systems is necessary for a reliable operation of the device. If possible, these calibrations should work automatically, and the results should be reliable without too much human supervision. Semi-automatic calibration procedures were developed for the LYCCA ToF detectors in section 4.1.3, and for the DSSSD detectors in section 4.1.4. A proper, absolute calibration of the CsI detectors is shown in section 4.1.6, which was never done before. A suggestion to extract more information about the mass and charge of the ions from the measured quantities by taking advantage of correlations is shown in 4.1.7. These developments are important in order to have any chance analyzing data from experiments with heavier ions, e.g. S429 experiment, performed in September 2012 (see table 4.1).

4.1.1 Overview of the detector system

In the following, a brief description of the LYCCA detector setup will be given. This description follows [23] and [55] where more details can be found about the LYCCA setup.

LYCCA consists of independent detectors: the ToF detector system, the E - ΔE wall and the target DSSSD detector. The drift distance of the ToF system is defined by two independent circular Saint-Gobain BC-420 plastic scintillation detectors with 270 mm diameter and 1 mm thickness, read out by 32 Hamamatsu R7400U photomultiplier tubes (PMTs) (figure 4.2 a). The detectors are labeled as ToF-start and ToF-stop in figure 4.1. One additional quenched Saint-Gobain BC422Q(0.2%) plastic scintillator read out by 12 Hamamatsu R9880-210 PMTs (figure 4.2 c) can be placed inside the secondary target chamber in order to measure the difference between incoming and outgoing velocity (ToF-target in figure 4.1). The PMTs are arranged around the scintillators to collect the scintillation light and produce time signals. The required time resolution of the ToF measurement is achieved by combining independent time measurements of the

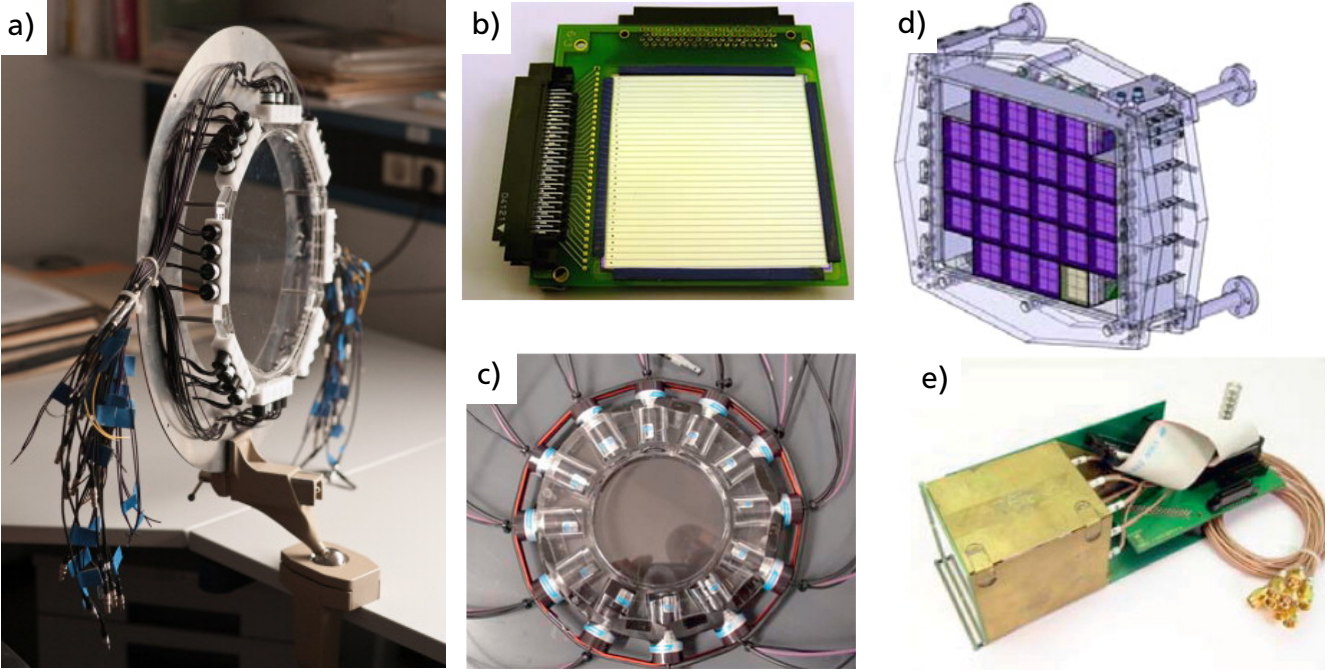


Figure 4.2.: a) ToF-Start / ToF-Stop detector, reprinted from [55], Copyright (2011) with permission from Elsevier. Target DSSSD b), ToF-Target detector, c) LYCCA wall assembly d) and LYCCA wall module e), reprinted from [23], Copyright (2013) with permission from Elsevier. The arrangement of these detectors can be seen in figure 4.1.

scintillation light from all N_{PMT} PMTs with a resolution of σ_{PMT} to get a combined time with a resolution of $\sigma_t = \sigma_{\text{PMT}} / \sqrt{N_{\text{PMT}}}$. It will be shown in section 4.1.3, how a good estimate of the ToF detector performance can be obtained. Before combining the individual timing measurements, the propagation time of the scintillation light from the impact point of the heavy ion to the PMT has to be taken into account. In section 4.1.3 it will be shown how to use this propagation time to extract additional position information from the LYCCA ToF detectors.

The E - ΔE wall (figure 4.2 d) is build out of E - ΔE modules that have one double-sided silicon strip detector (DSSSD) for ΔE and position measurement, and nine CsI:Tl scintillators for kinetic energy measurement in one housing (figure 4.2 e). The scintillators have a square-shaped front face area of $19.4 \times 19.4 \text{ mm}^2$ and two different thicknesses of 33 mm or 10 mm for different beam energies and are coupled with tapered light guides to $10.3 \times 10.3 \text{ mm}$ large silicon photodiodes for read out. In section 4.1.6, the process of correctly calibration these scintillators is described. DSSSD detectors are square-shaped with outer dimensions of $60.1 \times 60.1 \text{ mm}^2$ and a thickness of $300 - 320 \mu\text{m}$. Their active area is $58.5 \times 58.5 \text{ mm}^2$ large and divided into 32 strips on the p-side and the n-side. Strips on opposite sides are perpendicular to each other so that the ion-position can be determined from the intersection point of p-side and n-side strips. Section 4.1.4 describes a new calibration technique for DSSSDs that works automatically and is in many ways superior to the calibration with sources or a mono-energetic beam. In the experimental PreSPEC campaigns in 2012 and 2014, 16 LYCCA-wall modules were used. According to the LYCCA technical design report [56], the design goal of the detector system resolution is:

$$\frac{\Delta E_{\text{DSSSD}}}{E} = 0.5\% \text{ (FWHM)} \quad , \quad \frac{\Delta E_{\text{CsI}}}{E} = 0.5\% \text{ (FWHM)} \quad , \quad \Delta \text{ToF} < 50 \text{ ps (FWHM)} . \quad (4.1)$$

4.1.2 Principle of operation

For ion identification with the LYCCA device, A and Z have to be derived from measurements of total kinetic energy $E_{\text{kin}} = \Delta E_{\text{DSSSD}} + E_{\text{CsI}}$, the energy loss ΔE_{DSSSD} in the thin silicon detector of thickness Δx and the particle velocity β . The specific energy loss can be determined by $|dE/dx| = \Delta E_{\text{DSSSD}}/\Delta x$. The aim is to achieve for all these detectors an absolute calibration to be able to get a global identification of all detected isotopes. Absolute calibration means in this context the conversion from detected signal amplitudes to physical quantities, for example in the case of the CsI wall detectors an energy value in units of MeV.

Determining Z from β and ΔE_{DSSSD} measurements

The mean energy loss of charged particles in matter due to the interaction with target electrons is described by the Bethe formula [57]

$$\left\langle -\frac{dE}{dx} \right\rangle = K \frac{Z_{\text{tar}}}{A_{\text{tar}}} \frac{Z^2}{\beta^2} \left[\frac{1}{2} \ln \frac{2 m_e c^2 \beta^2 \gamma^2 W_{\text{max}}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right], \quad (4.2)$$

and depends on the target mass A_{tar} and charge Z_{tar} number, the mean ionization potential I , and the particle charge number Z and velocity $\beta = v/c$. The maximum energy transfer for particles with mass M in a single collision W_{max} is given by

$$W_{\text{max}} = \frac{2 m_e c^2 \beta^2 \gamma^2}{1 + 2 \gamma m_e/M + (m_e/M)^2}. \quad (4.3)$$

The quantity $K = 4\pi N_A r_e^2 m_e c^2$ is composed of fundamental constants, including the classical electron radius $r_e = e^2/(4\pi\epsilon_0 m_e c^2)$. With a value of $K = 0.307 \text{ MeV mol}^{-1} \text{ cm}^2$ and A in units of g mol^{-1} , the result of (4.2) will be in units of $\text{MeV}/(\text{g}/\text{cm}^2)$. The term $\delta(\beta\gamma)$ is a correction for very high energies that is not considered here.

A more accurate description of the energy loss is available in the ATIMA computer program [58]. It is valid over a larger energy range and also predicts angular and energy straggling effects. For the Bethe formula (4.2) and the ATIMA program, the energy loss was calculated for ^{80}Kr particles in silicon over a large energy range. The result is shown in figure 4.3. In order to determine Z , equation (4.2) can be approximated by a power law and an empirical formula for the measured energy loss can be obtained:

$$\Delta E = \left| \frac{dE}{dx} \right| \Delta x \propto \frac{Z^2}{f(\beta(E_{\text{kin}}, m))} \approx \chi \frac{Z^2}{\beta^\kappa} \quad (4.4)$$

with two empirically determined constants χ and κ . The energy loss as described by equation (4.4) compares very well to the ATIMA values inside an interval $\beta \in [0.3, 0.8]$, as can be seen in figure 4.3. Values for κ and χ can be obtained by a nonlinear fit of equation (4.4) to the ATIMA calculations, and the results are: $\kappa = 1.535(4)$ and $\chi = 1.547(6) \cdot 10^{-3} \frac{\text{MeV}}{\text{g}/\text{cm}^2}$. The chosen interval, where the power law approximation is good, is sufficiently large for most types of experiments that use LYCCA for particle identification. Equation (4.4) can be rearranged to get a formula for Z

$$Z = \sqrt{\Delta E_{\text{DSSSD}} \frac{\beta^\kappa}{\chi}} \quad (4.5)$$

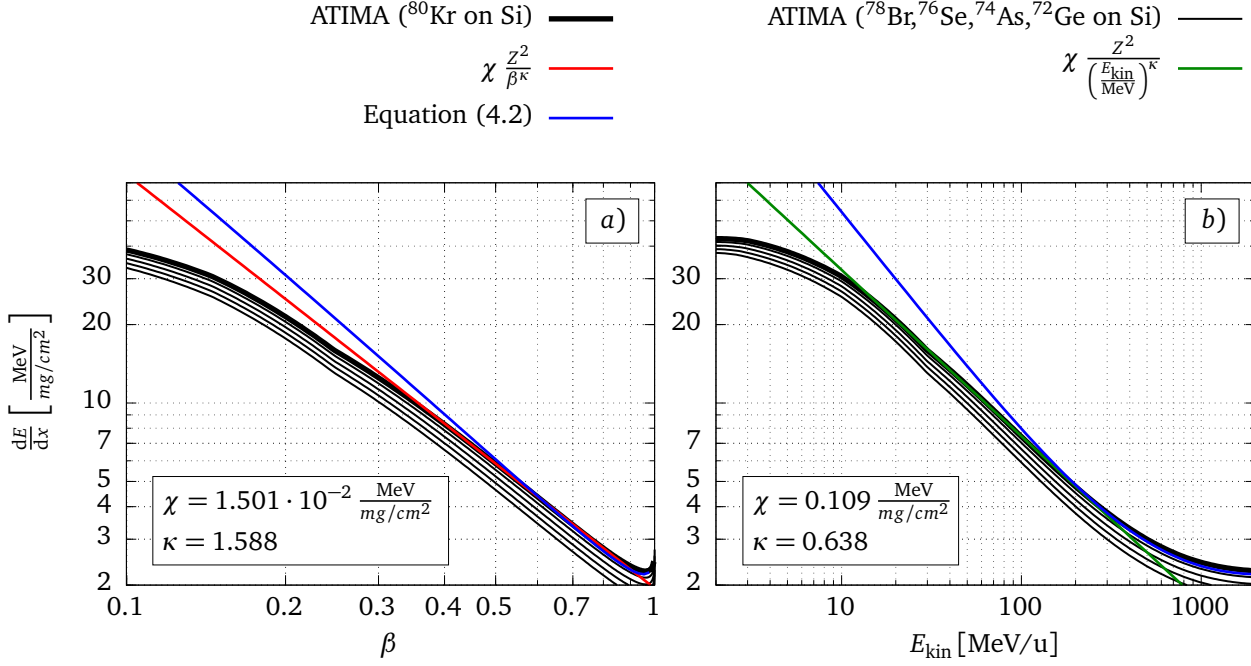


Figure 4.3.: The thick, black line shows the loss of ^{80}Kr particles in silicon as calculated with the ATIMA program [58]. Thin, black lines are the ATIMA result for lighter isotopes. Panel a) shows the velocity dependence, while panel b) shows the kinetic energy dependence. The blue line is obtained with the Bethe formula (4.2). Both, ATIMA and the Bethe formula are in good agreement above 150 MeV/u. Over a large range of kinetic energies (≈ 15 MeV/u to ≈ 200 MeV/u), the ATIMA energy loss has the form of a power law (red line for the β dependence and green line for the E_{kin} dependence).

With equation (4.5), one can directly obtain the nuclear charge number Z of the particle in an event. In figure 4.4 a) a correlation plot is shown for β and ΔE_{DSSSD} from experimental data, together with equation (4.4) for different Z . Note that the values of κ and χ that fit the experimental data are slightly different than the ones fitted to the ATIMA calculation.

Determining A from β and E_{kin} measurements

For mass identification, the relation between kinetic energy, mass and velocity can be directly used:

$$E_{\text{kin}} = (\gamma(\beta) - 1) m c^2 \xrightarrow{m=\text{Au}} A = \frac{E_{\text{kin}}}{(\gamma(\beta) - 1) u c^2}. \quad (4.6)$$

This requires a correct calibration of all CsI detectors, which is challenging because CsI has a Z dependent, non-linear light yield for heavy ions [59], [60]. Section 4.1.6 explains how the calibration of the CsI detectors was done. Figure 4.4 b) shows a correlation plot of β and E_{kin} , where the different masses can be seen. After correct calibration, the lines in the data are consistent with the description by the kinetic energy formula (4.6).

Finally, obtained values for A and Z can be combined into a LYCCA particle identification plot as can be seen in figure 4.5. The distribution of isotopes observed in experiment S424 reached from the primary beam ^{80}Kr , down to some aluminum isotopes around mass $A = 27$.

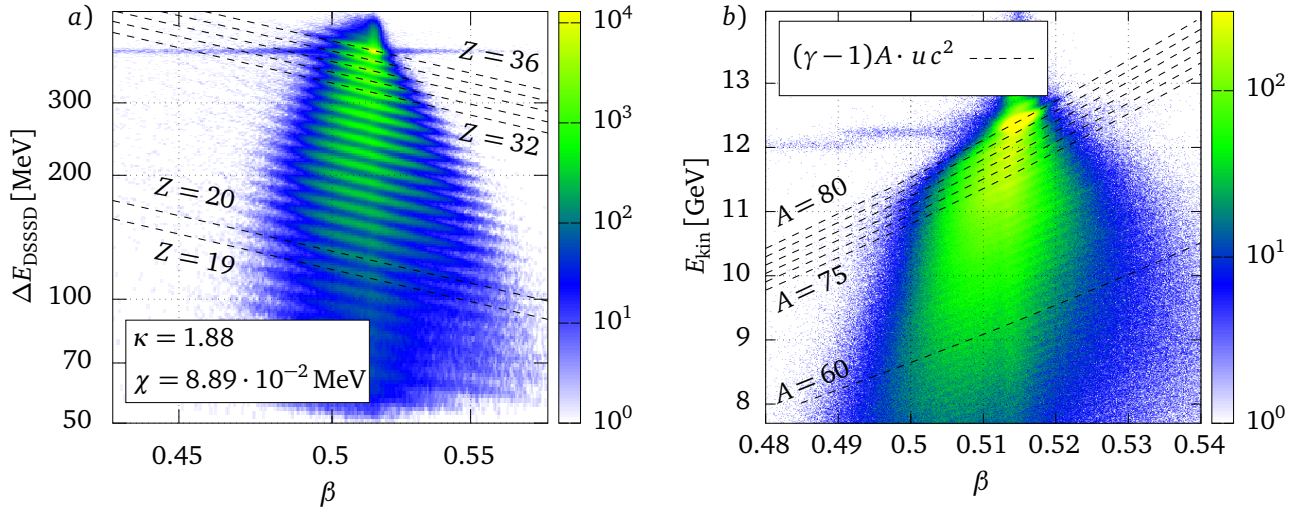


Figure 4.4.: Panel a): ΔE - β histogram with logarithmic axis and color scaling, showing measured data from the calibrated LYCCA detectors. The data is from runs 481-508 of the S424 experiment in 2012 with a particle gamma trigger condition. It is possible to identify different Z values on distinct lines that approximately follow power laws as in equation (4.4) represented by dashed lines for different values of Z . Panel b): Histogram of β - E_{kin} data. Dashed lines represent the theoretical velocity dependence of E_{kin} . Distinct lines in the data are difficult to see because of the limited resolution of the LYCCA detectors.

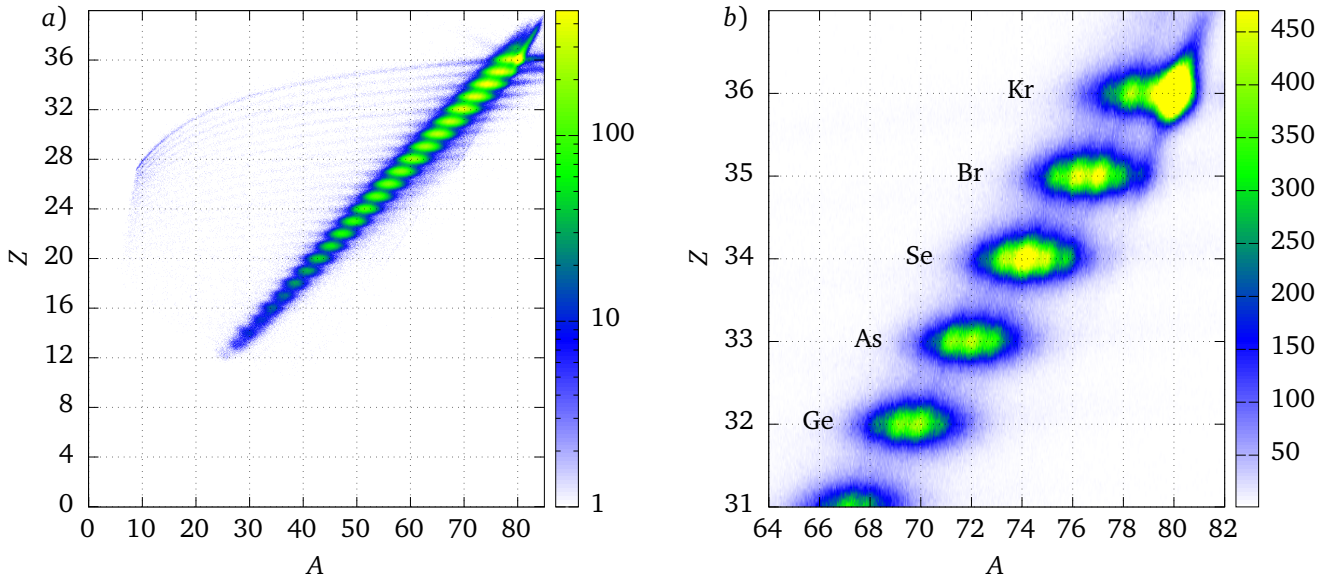
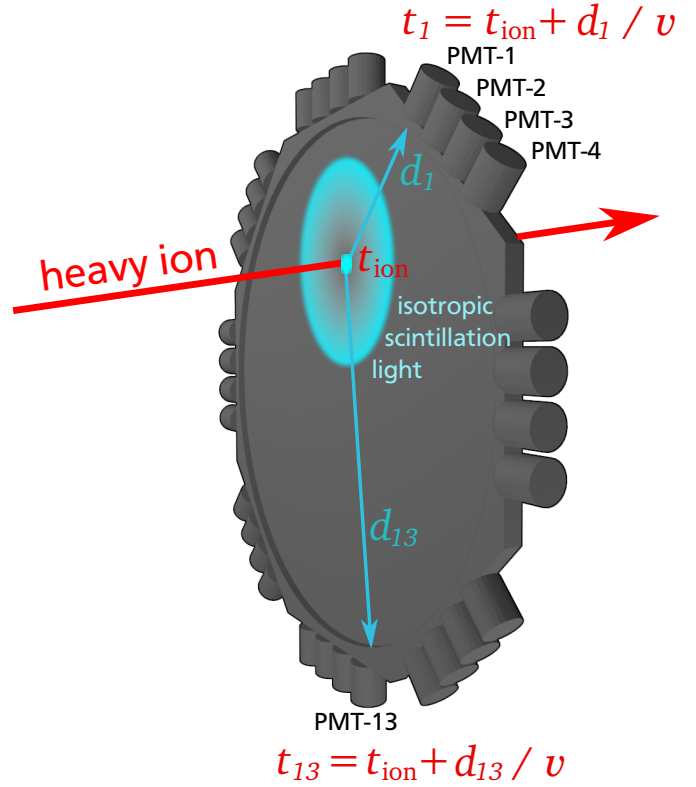


Figure 4.5.: a) Final LYCCA particle identification plot. b) Zoom into the region of the incoming isotope ^{80}Kr .

Figure 4.6.: Principle of multiple measurements of the same incident heavy ion event on a circular sheet of plastic scintillator. The drawing represents the layout of the LYCCA ToF-start or ToF-stop detector. 32 PMTs surround the scintillator in eight groups of four. Heavy ion impacts generate scintillation light which is guided through the scintillator to the PMTs. In the drawing, the impact position is closer to PMT-1 and further away from PMT-13. The respective signals will have a different time offset (t_1 for PMT-1, and t_{13} for PMT-13) that has to be corrected, before the measured time values are averaged to obtain the ion impact time t_{ion} .



4.1.3 Time-of-flight detectors

LYCCA performs a time-of-flight measurement using two or three circular scintillation detectors: the first one is located 1419(5)mm before the target position, the second one is located 115(5)mm before the target, and the third one is placed in front of the ΔE -E wall, 3597(5)mm after the target. The second one was inside the target chamber and was removed in some of the experiments to reduce the amount of matter in the target chamber. In this section, it will be discussed how to calibrate the time-of-flight system. In a first step, all PMTs in each detector have to be calibrated for the ion-position dependent light propagation time from the ion impact position to the PMT. In a second step, an offset calibration for the three detectors is needed to calculate the particle velocity from the three measured times.

Intrinsic calibration of time-of-flight detectors

LYCCA time-of-flight detectors achieve a high timing accuracy well below 30 ps (FWHM) by averaging multiple timing measurements of the same event. In case of the start and stop detectors, there are 32 time signals per event that are combined to a single value for the time t_{ion} when the ion penetrated the detector. Figure 4.6 shows the principle of operation of such a detector. Optical photons created at the point \vec{r}_{ion} , where the ion hits the detector need to propagate to the PMT. The propagation time is assumed to be a linear function of the distance $d_i = |\vec{r}_{\text{ion}} - \vec{r}_{\text{PMT},i}|$ between the ion position and the position $\vec{r}_{\text{PMT},i}$ of PMT- i . It is further assumed that the light propagates with the effective velocity v inside the scintillator. The problem is straight forward if r_{ion} and v are known. The propagation time can be corrected by

$$t_{i,\text{cor.}} = t_i - (o_i + d_i/v), \quad (4.7)$$

where the time offset o_i can be used to align all corrected times to zero. Given N_{PMT} corrected time measurements, the best estimate for t_{ion} is given by the average of all these times:

$$t_{\text{ion}} = \frac{1}{N} \sum_{i=0}^N t_{i,\text{cor.}} \quad (4.8)$$

Effective velocity and light propagation time correction

The total distance traveled by the light is longer than the distance d_i between the ion impact point and the PMT- i , because light inside the scintillator is conducted by total reflection at the inner surface. BC-420 has a refraction index of $n = 1.58$ [61] which corresponds to a critical angle of $\varphi_{\text{crit.}} = 39.27^\circ$, measured to the scintillator surface normal. For photons emitted at $\varphi_{\text{crit.}}$, the way to the scintillator is longer by a factor of $1/\sin(\varphi_{\text{crit.}}) = 1.58$. The additional distance can be treated as a decrease in the effective velocity along the two dimensional sheet of scintillator material. Depending on the angle at which most of the light will reach the PMT, the effective velocity can be between $\sin(\varphi_{\text{crit.}})c/n \leq v \leq c/n$, in numbers $0.4c \leq v \leq 0.63c$. In practice, it is best to determine v from the experimental data by plotting the PMT time over the PMT-ion distance. The ion position can be obtained from the LYCCA target and wall DSSSD tracking in case of the ToF-stop detector, or the FRS TPC and the LYCCA target DSSSD tracking in case of the ToF-start detector. For the ToF-target detector, one can take directly the target DSSSD position without introducing large errors because the detectors are only separated by a few cm.

An additional problem arises from the fact that the TDCs (CAEN v1290 [62]) are accurate down to 25 ps only for *differences* of two synchronized channels. For a single time measurement, all channels have a correlated jitter. In order to create a correlation plot of the propagation distance and time, a synchronized reference time has to be subtracted in order to remove the jitter and to determine the value of v and o_i . In [55], the PMTs were treated in pairs, each one serving as synchronized reference for the other. However, with the value of t_{ion} , a much more precise reference is available. The downside is, that before v and o_i were determined, the value of t_{ion} cannot be calculated. It turned out that an initial guess of $o_i = 0$ and $v = c/3$ works good enough to create correlation plots of d_i and t_i . From these correlation plots, better values of o_i and v can be extracted. This procedure can be repeated a few times until the resulting calibration does not improve any further. Some correlation plots for the ToF-stop detector can be seen in figure 4.7. The effective velocity v has different values for different PMTs. For the ToF-start detector the range of measured values for v_i was $[0.34c, 0.51c]$, for the ToF-target detector it was $[0.26c, 0.78c]$ and for the ToF-stop detector it was $[0.28c, 0.54c]$. These intervals clearly contain values that are outside the interval of $[0.4c, 0.63c]$ as calculated before. This inconsistency was not further investigated, because the ToF detector performance reached its specifications.

The arrival time t_{ion} of the ion is calculated as the non-weighted average of all PMT signals that have a resolution better than a given acceptance threshold, using equation (4.8). If the PMT resolution is known, it would be possible to use the weighted average to calculate t_{ion} . In this way, the introduction of an acceptance threshold is not needed. However, the iterative procedure to determine o_i and v_i was unstable if weighted averages were used. Also, the resulting performance is not significantly worse if non-weighted averages are used as is shown figure 4.8 of the next section.

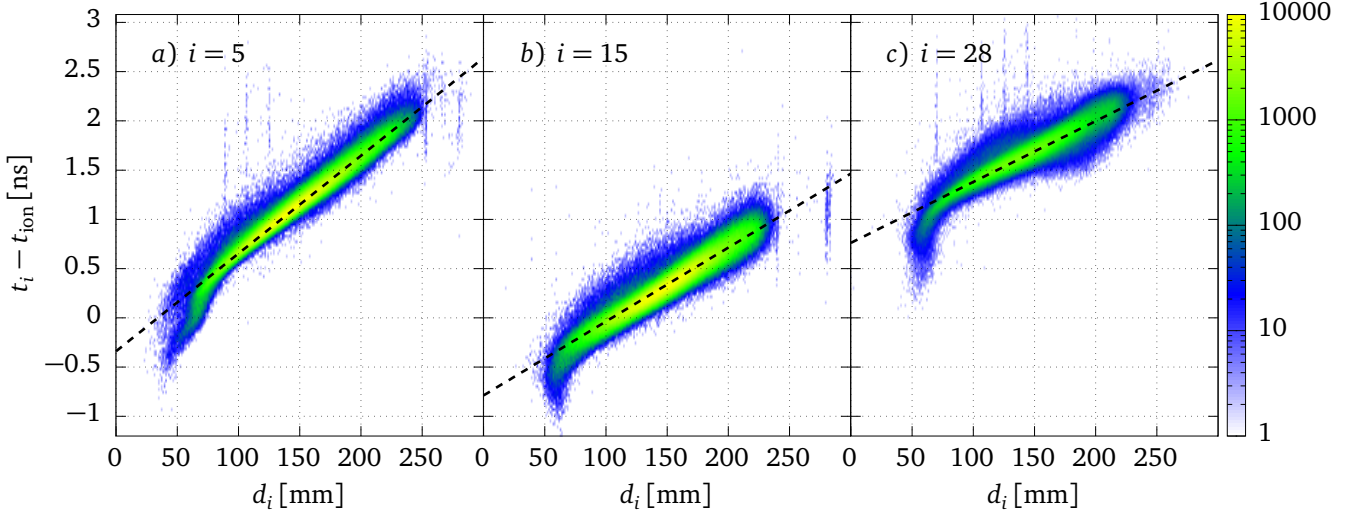


Figure 4.7.: Examples of correlation plot of distance between PMT and ion impact point d_i and the time delay after the ion impact $t_i - t_{\text{ion}}$ for PMT-5, PMT-15 and PMT-28. Dashed lines represent the extracted o_i and v_i .

Intrinsic determination of the time resolution

One outstanding feature of the LYCCA ToF detectors is their ability to determine the time resolution without any external time reference, using the following procedure. First, the time resolution $\Delta t_{i,\text{cor.}}$ of all PMTs will be calculated using t_{ion} as time reference. Then, the time resolution of t_{ion} can be calculated from all $\Delta t_{i,\text{cor.}}$. The uncertainty of t_{ion} is much smaller ($\Delta t_{\text{ion}} \approx 30 \text{ ps}$) than the uncertainty of $t_{i,\text{cor.}}$ ($\Delta t_{i,\text{cor.}} \approx 130 \text{ ps}$), and can be ignored in the calculation. The signal amplitude in the PMT depends on the Z of the detected ion and therefore influences the achievable time resolution which depends on the noise σ , amplitude A and rise-time t_{rise} of the signal

$$\Delta t_{\text{ion}} \propto \sigma \frac{t_{\text{rise}}}{A}. \quad (4.9)$$

This means, that the best time resolution can be expected for the highest Z ions. The time resolution presented here was done with a $Z = 36$ gate in the LYCCA particle identification and is valid only for this case. The amplitude of the scintillation light depends on Z^2 , therefore the expected time resolution for different elements varies with Z as $\Delta t_{\text{ion}} \propto 1/Z^2$.

From the $t_{i,\text{cor.}}$ distributions of each PMT (see figure 4.8 a)), the variance was determined. The resolution of the full detector is calculated with a Monte Carlo method by drawing random samples from Gaussian distributions with the measured variances and calculating their non-weighted average. The detector resolution is the width of the resulting distribution of average values. These numbers were also compared with the achievable resolution using weighted averages. Figure 4.8 b) summarizes the time resolution information for each of the three detectors and shows $t_{i,\text{cor.}}$ distributions for all PMTs of the ToF-start detector.

The resulting values are compared with the design value of $\Delta t_{\text{design}} \leq 35.4 \text{ ps} = 50 \text{ ps}/\sqrt{2}$ in Tab. 4.2. The design value was calculated from the desired ToF resolution of 50 ps and the assumption that ToF is measured with two identical detectors. ToF-start and ToF-stop are

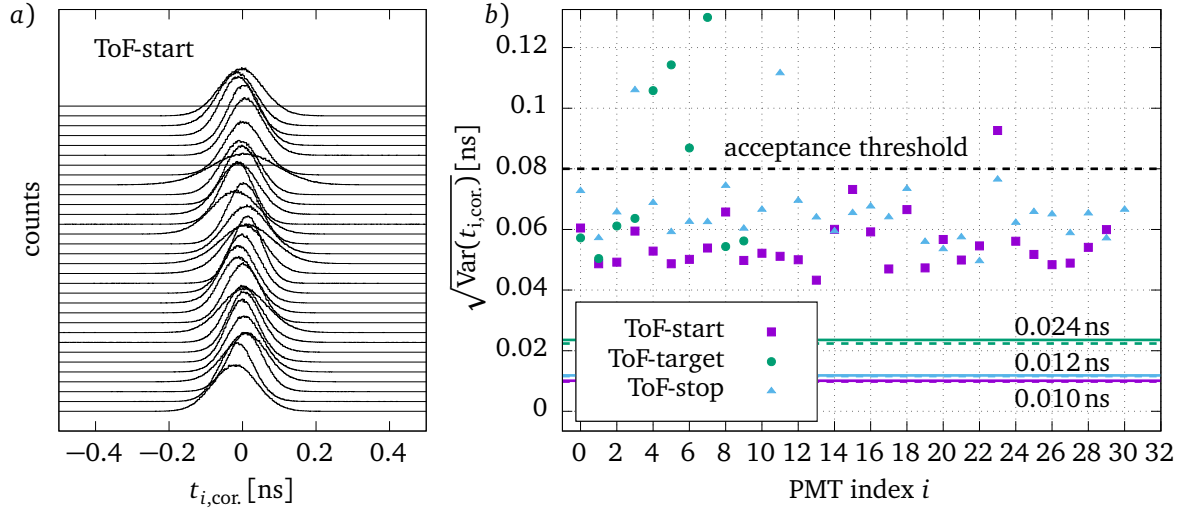


Figure 4.8.: Left: Histograms of $t_{i,cor.}$ for all PMTs of the LYCCA ToF-start detector with a gate on $Z = 36$ and a ^{80}Kr beam. It can be seen from the width of the distributions, that some PMTs perform significantly worse than the rest. Right: Standard deviation of $t_{i,cor.}$ for each PMT in each detector. Different Detectors are shown in different colors. The black dashed line is at the acceptance threshold above which a PMT was not considered in the calculation of $t_{ion.}$. Colored solid lines show the resulting standard deviation of the non-weighted average of the accepted PMTs. Colored dashed lines show the possible standard deviation if a weighted average would have been calculated using all PMTs.

detector	Δt_{design} (FWHM)	$\Delta t_{\text{Exp.}}$ (FWHM)
ToF-start	$\leq 35.4 \text{ ps}$	23.5 ps
ToF-target	$\leq 35.4 \text{ ps}$	56.4 ps
ToF-stop	$\leq 35.4 \text{ ps}$	28.2 ps

Table 4.2.: Summary of the LYCCA ToF detector performance. ToF-start and ToF-stop are clearly within their specifications. ToF-target is about a factor of two worse than the two big detectors.

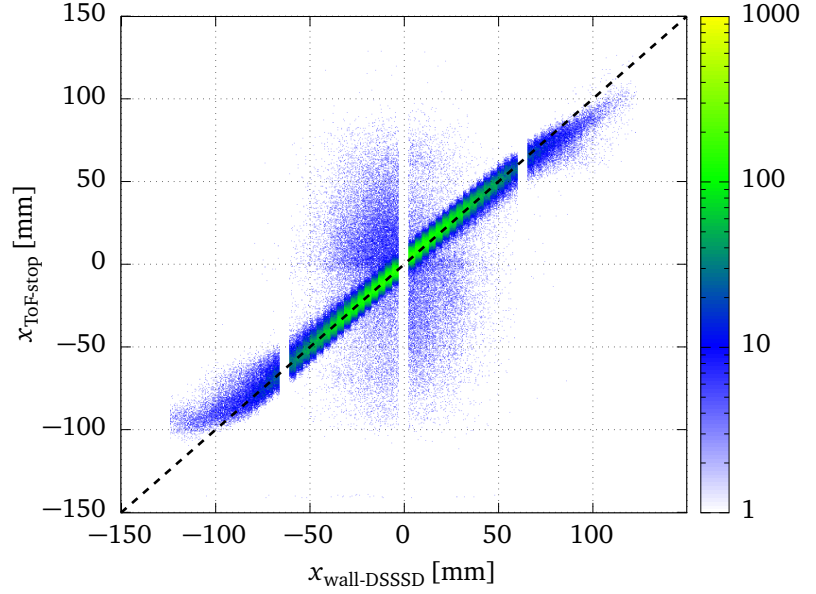
clearly better than their design values, but ToF-target is worse by roughly a factor of two. However, ToF-target could only use half of the PMTs, either because of complete failures, or the PMT time resolution was above the acceptance threshold. If all twelve ToF-target PMTs could have been used, performance would have been better by a factor of $\approx \sqrt{2}$ better, resulting in $\Delta t_{\text{ToF-target}, 12 \text{ PMT}} = 39.6 \text{ ps}$ which would almost match its design value.

Position sensitivity of the time-of-flight detectors

Having all PMT signals aligned and calibrated as shown in the previous paragraph, it is possible to extract particle position information from the time signals. This technique relies on the fact that the $t_{i,cor.}$ distributions are centered around zero if the particle position is correct. If the particle position is unknown, a particle position x, y can be assumed and the variance

$$\text{Var}_{x,y}(\{t_{i,cor.}\}) = \langle t_{i,cor.}^2 \rangle - \langle t_{i,cor.} \rangle^2 \quad (4.10)$$

Figure 4.9.: Correlation of the x position of the particles, measured with the LYCCA ToF-stop detector and the LYCCA wall-DSSSD detectors. The gaps in x_{DSSSD} correspond to the area between two adjacent DSSSDs.



of the set $\{t_{i,\text{cor.}}\}$ of N_{PMT} time signals from that event. This variance depends on x and y and can be interpreted as likelihood function of the particle position. The particle position can be obtained by taking a reasonable starting point, e.g. x_0, y_0 at the center of the detector, and move the coordinates towards the minimum of $\text{Var}_{x,y}$ with a certain step size h . The parameter h was initially chosen to be $h = r_{\text{sctl.}}/7$, where $r_{\text{sctl.}}$ is the radius of the scintillation disk. When the minimum is found, the step size can be divided by two and the minimum can be searched again. Repeating this procedure until the step size is below a certain threshold (2 mm in this case). The values of x and y can then be taken as the position where the ion penetrated the scintillator. This is a novel technique to obtain particle position information and was published in [63].

In order to estimate the achieved position resolution, the particle position from the ToF-stop detector can be compared with the position from the wall-DSSSD detectors as reference. The correlation in x -position determined from wall-DSSSD and ToF-stop can be seen in figure 4.9. This was done in [63] and after careful tuning of all parameters for a single run, a position resolution of $\Delta x = 6.7 \text{ mm}$ and $\Delta y = 5.1 \text{ mm}$ could be achieved. The position resolution is worse than the width of wall-DSSSD strips, which is $2 \times 1.8 \text{ mm} = 3.6 \text{ mm}$, but not even by a factor of two. Using this technique, detectors of similar type could be used in applications where both, position and extremely precise time measurements are needed. Using recent silicon PMTs, such a detector could be much cheaper than a similar sized DSSSD array. The quality of the position determination depends on the quality of the intrinsic calibration of the detector. For that reason, a correlation plot as shown in Fig 4.9 can be used as a tool for ToF-detector diagnostics during an experiment.

Calibration of the velocity measurement

From the time measurements at different positions, the outgoing particle velocity can be calculated using the simple formula

$$\beta_{\text{out}} = \frac{z_{\text{stop}} - z_{\text{target}}}{(t_{\text{stop}} - t_{\text{target}})c}, \quad (4.11)$$

where the quantities z and t refer to positions along the beam line and the times measured at these positions by the respective ToF-target and ToF-stop detectors. The positions of ToF-target and ToF-stop detectors are known on an absolute scale. The time information is subject to some unknown influences, e.g. processing time of the front-end electronics and discriminator thresholds. Therefore, the measured time signals have to be corrected by a fixed time offset that has to be determined before equation (4.11) can be applied. The offset can be found by using information from the γ -ray spectrometer. In the Doppler correction of the γ -ray data, the velocity β_{out} at which the radiation was emitted from the outgoing particle is required. The value of β_{out} can either be taken event by event from the LYCCA ToF measurement, or by assuming a mean particle velocity $\langle\beta_{\text{out}}\rangle$ for all particles. The energy of the emitted γ -rays from the decay of the 2_1^+ state in ^{80}Kr is 616.6 keV. The mean particle velocity $\langle\beta_{\text{out}}\rangle$ can be chosen such, that the energy after Doppler correction is correct. This value can be used to tune the ToF offset such that it is in agreement with the mean value of the velocity from the LYCCA ToF system. The resulting LYCCA velocity calibration is good enough for global particle identification with LYCCA.

The resolution of the ToF-target detector was inferior to the other two timing detectors. To achieve a better resolution for β it is desirable to use the information from the ToF-start detector and measure the longer flight distance $x_{\text{stop}} - x_{\text{start}}$. Doing so, introduces a systematic error, because the particle velocity before the target chamber is higher than the velocity after the target chamber. If an equation similar to (4.11) is used

$$\beta_{\text{eff.}} = \frac{z_{\text{stop}} - z_{\text{start}}}{(t_{\text{stop}} - t_{\text{start}})c}, \quad (4.12)$$

the result is effective velocity between the two outer timing detectors, but not β_{out} . The best possible way to obtain the velocity after the target from the three times t_{start} , t_{target} , and t_{stop} would involve a model of the stopping power in the target area and a fit to the most likely velocity before and after the target that also agrees with the observed scattering angle in the target area. This was not done in the present work, but instead, an empirical correction was applied. First, the time offset for t_{start} was adjusted such that the peaks in the distributions of β_{out} and $\beta_{\text{eff.}}$ are at the same value. Using that offset would result in a wrong incoming particle velocity β_{in} . Then, an empirical correction was applied by comparing the correct but low resolution β_{out} from equation (4.11) with the more accurate but not correct $\beta_{\text{eff.}}$ from equation (4.12). The correction can be written as

$$\beta_{\text{eff.,cor.}} = \begin{cases} (\beta_{\text{eff.}} - \beta_0) \cdot \alpha_{<} + \beta_0 & \text{for } \beta_{\text{eff.}} \leq \beta_0, \quad \alpha_{<} = 1.29 \\ (\beta_{\text{eff.}} - \beta_0) \cdot \alpha_{>} + \beta_0 & \text{for } \beta_{\text{eff.}} > \beta_0, \quad \alpha_{>} = 1.08 \end{cases} \quad \text{with } \beta_0 = 0.513, \quad (4.13)$$

and is such that the correlation plot of both quantities (see figure 4.10) is along the $\beta_{\text{out}} = \beta_{\text{eff.,cor.}}$ diagonal.

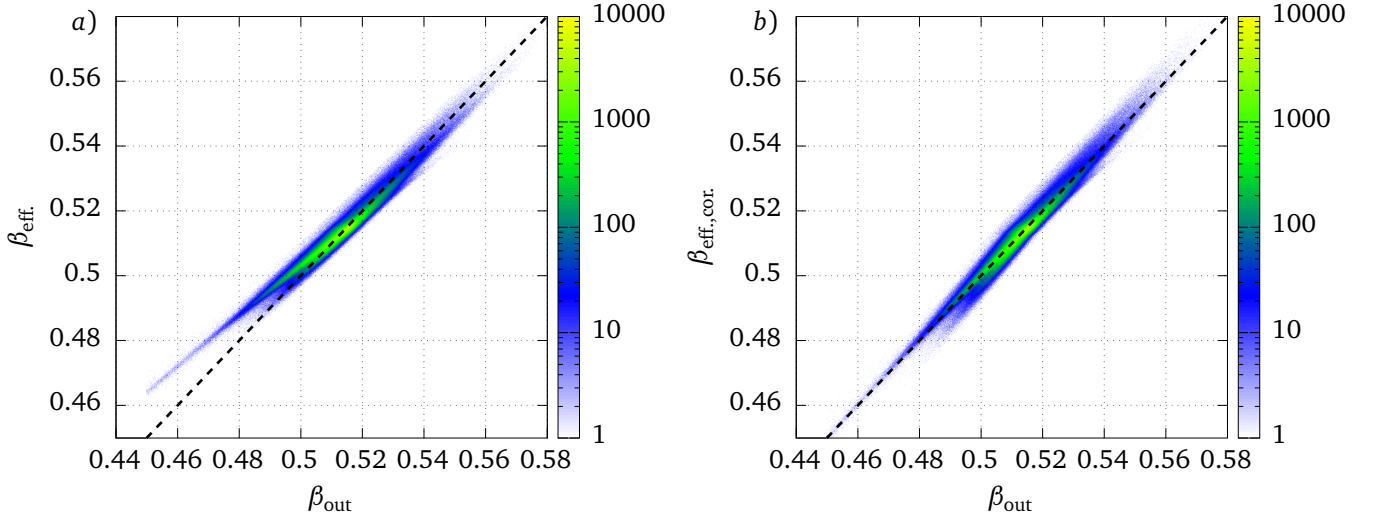


Figure 4.10.: Panel a) shows a correlation plot of β_{out} and β_{eff} . In panel b), an empirical correction was done that effectively corrects the shape of this plot to be diagonal, shown as a dashed line.

Finally, the weighted average of β_{out} and $\beta_{\text{eff,cor}}$ is calculated and is used as the particle velocity β . The β distribution of heavy ions as measured in the S424 experiment for the secondary fragmentation of ^{80}Kr on a beryllium target is shown in figure 4.8. The observed peak width is an upper limit for the LYCCA β resolution: $\Delta\beta/\beta < 0.53\%$ (FWHM), i.e. the achieved value is at least as good as the design value.

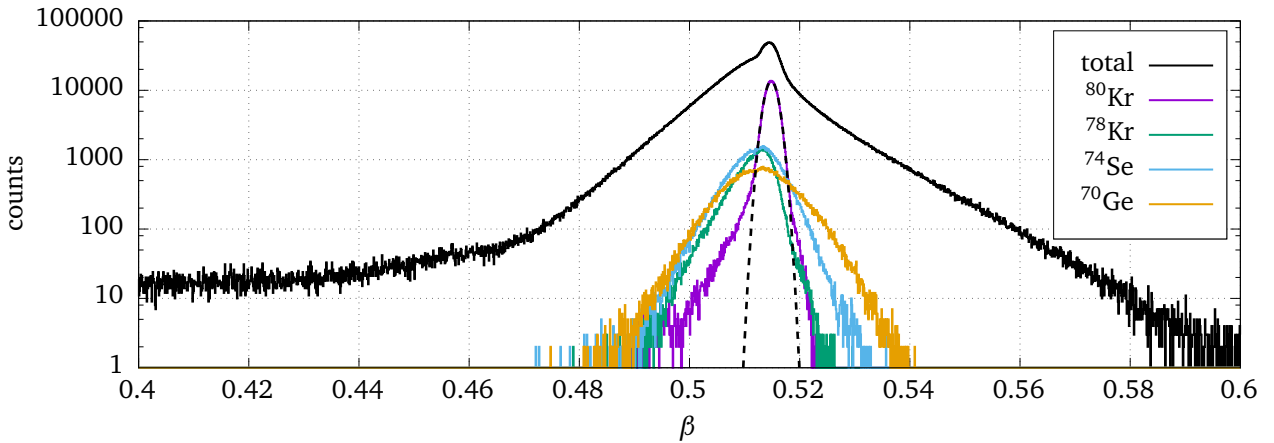


Figure 4.11.: Velocity distribution histograms for different isotopes. It can be seen that the more exotic reactions yield broader velocity distributions. The dashed line shows a Gaussian fit to the velocity distribution of the primary beam. The width is an upper limit of the resolution of the LYCCA velocity measurement: $\Delta\beta/\beta < 0.53\%$ (FWHM)

4.1.4 DSSSD calibration

The DSSSDs of LYCCA have two tasks: spatial tracking and energy-loss measurement of the ions. The position of an ion in the DSSSD is determined by the segment that was hit. This is different than the position determination in the CATE Si-detectors, which used non-segmented Si-detectors that were covered with a resistive layer and signal read-out on the four corners of the detector [51]. The position was determined by the amplitude ratio of signals at the corners of the detector.

Compared to the CATE Si-detectors, the position determination in the LYCCA DSSSDs is easier, because it is determined by the strip number on the p-side and n-side. For energy-loss measurements, the high segmentation is not helpful, because each segment needs to be calibrated. Another disadvantage of the high segmentation is the low active area of a single strip. This leads to long measurement times if traditional calibration techniques with known sources or known beam energies are used. These techniques rely on a sufficient number of events in each single analog channel, in order to determine one (or more) peak positions that determine the gain of the respective channel. This procedure was successfully done for LYCCA (see [64]), but considerable amounts of expensive beam-time have been spent for the calibration. An additional disadvantage of the traditional calibration technique is the amount of manual work required to do it. If there are only few counts in the calibration spectrum, the peak positions have to be verified by humans, even if automatic peak finding algorithms are used.

Automatic intrinsic calibration of the LYCCA DSSSD detectors

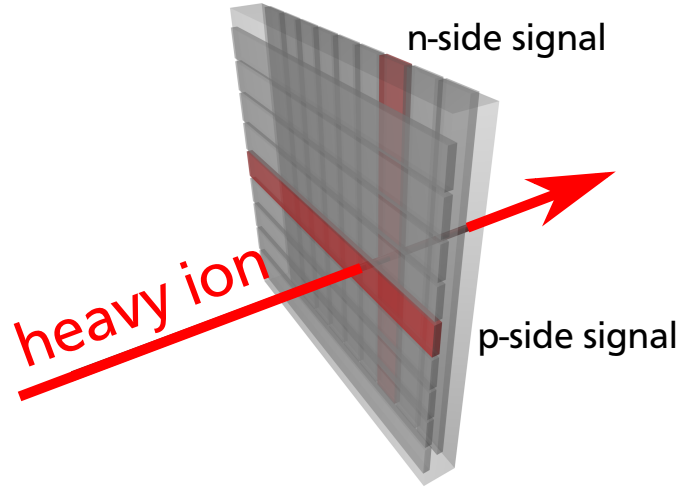
This section presents a technique that was developed within this work, which allows to achieve an intrinsic calibration of a DSSSD detectors in a reliable way, using production data without any particular energy distribution. Intrinsic calibration means that an event with a given energy deposition will result in the same detected amplitude, regardless which pixel was hit. The term gain-matching is also used in this context.

The method uses the correlation between signal amplitudes measured on both sides of the detector. This correlation is always present, because both signals are originating from the same particle. This underlying principle, the determination of calibration coefficients from p-side and n-side correlations, can be applied to all kinds of DSSSDs used in nuclear or particle physics. However, the presented implementation is very simple and relies on a relatively large strip width, i.e. this implementation cannot be used for micro-strip detector calibration. The simple implementation turned out to be sufficient for achieving good results with LYCCA DSSSDs. This technique was published in [65], and the following text is a slightly modified version of that reference. A different implementation of the same principle, which would also work for micro-strip detectors, was done by Stahl [66].

Underlying principle

Each ion that passes through the bulk material of a DSSSD causes charge carriers to be created along its trajectory. These charge carriers drift towards the electrodes along the electrostatic field lines and create electrical signals. As in any segmented detector, the measured signal amplitudes of a single segment (strip) of the DSSSD depend on the relative position of the strip to the hit position on the detector. The largest signal will appear on the strip that was closest to

Figure 4.12.: Schematic drawing of a square-shaped DSSSD detector, being hit by a heavy ion. While passing through the bulk silicon, energy is transferred to electrons, which are drawn by the bias voltage towards the segmented electrodes in the shape of perpendicular strips (indicated in red). The hit position of the ion can be estimated by the intersection point of the p-side and n-side strips. The signal height in both strips is correlated, because they are collecting holes and electrons of the same event.



the particle impact. Cross-talk between different channels in the front-end electronic can modify the signal amplitudes of all channels. The presented algorithm neglects cross-talk and assumes that there are events in the data set, where the deposited energy

$$E = s_i A_i, \quad (4.14)$$

in the detector is proportional to a signal in channel i of amplitude A_i , and that all other channel amplitudes are negligible, i.e. single-strip hits:

$$A_i \gg A_j \quad \text{for } i \neq j. \quad (4.15)$$

The slope factor s_i is the only calibration coefficient for a given channel i . In section 4.1.4 the formalism will be generalized (4.14) to include an offset in the energy dependence. Note that the limitation to events that fulfill equation (4.15) is due to the chosen implementation. In principle, the possibility to obtain an intrinsic calibration of a DSSSD by using the correlation between p-side and n-side does not demand the exclusion of any events.

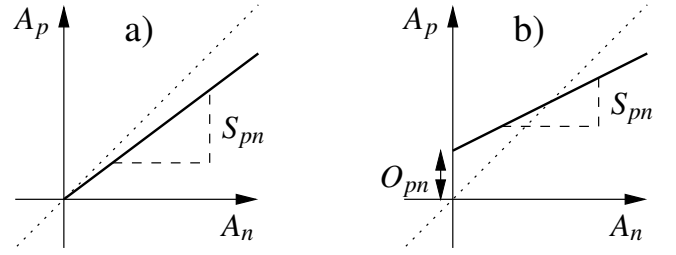
Given a DSSSD with N_p and N_n strips on the p-side and n-side, respectively, each event that is registered in a given pixel will create a signal with amplitude A_p in the strip number p on the p-side and a signal with amplitude A_n in strip number n on the n-side ($n, p = 1 \dots N_{n,p}$). Assuming that both strips measure the same deposited energy E in the active area of the detector, one can write

$$E_p = s_p A_p, \quad E_n = s_n A_n \quad \text{and} \quad E_p = E_n = E, \quad (4.16)$$

with s_p and s_n being the calibration coefficients for the p -th p-side strip and the n -th n-side strip, respectively. The correlation between the amplitudes on both sides allows to assume that A_p and A_n are linearly related. Under the assumption that equation (4.14) is valid, i.e. the calibration has no offset, and the relation between the two measured amplitudes is also linear without offset

$$A_p = S_{pn} A_n. \quad (4.17)$$

Figure 4.13.: The relation between A_p and A_n for one single pixel of the detector is shown as thick, solid line with slope S_{pn} . In general, this line does not coincide with the diagonal (dotted line) because both channels have different, linear dependence on the deposited energy. In a) no offset is present, while in b) an offset O_{pn} is allowed (see section 4.1.4).



The slope $S_{pn} = A_p/A_n$ of this line can be experimentally determined for each pixel of the detector, based on a given set of measured events where the particle hit the respective pixel. A schematic representation of these relations can be seen in figure 4.13.

For each pixel, this slope S_{pn} can be determined from a correlation plot of the amplitudes A_p and A_n . Figure 4.14 a) shows such a plot for all pixels of the LYCCA target DSSSD, where all different slopes are superimposed. The correlation of A_p and A_n is the reason why no monoenergetic source is needed: each single event contains information about the calibration factors s_p and s_n of both strips that were hit. After determining S_{pn} -values for all of the $N_p N_n$ pixels of the detector, these values can be used to get a set of $N_p + N_n$ calibration coefficients $\{s_p, s_n\}$ that best reproduces the set of measured $\{S_{pn}\}$. Both sets are related by

$$S_{pn} = \frac{s_n}{s_p}, \quad (4.18)$$

which follows from Eqs. (4.16,4.17). One way of finding a set of $2N$ calibration parameters $\{s_p, s_n\}$ is to minimize the following expression

$$\chi^2 = \sum_{p,n} \left(\frac{S_{pn} - \frac{s_n}{s_p}}{\Delta S_{pn}} \right)^2, \quad (4.19)$$

where ΔS_{pn} is the uncertainty for the pixel slopes S_{pn} as obtained from the measured data. The set of calibration parameters that minimize (4.19), also fulfill the condition (4.16) and therefore represent the best set of intrinsic calibration coefficients for a given input data set, if the simplifying assumptions are valid.

The presented method works only, if some conditions are fulfilled: Correlations between p-side and n-side signals have to be present, i.e. both sides have to be read by the data acquisition and there must be intersection point of the segments of both sides. A sufficient amount of single-strip events have to be present, where equation (4.15) is valid. Events with inter-strip hits on one or both sides will contribute to the background and should be excluded from the calibration procedure.

Implementation

The presented implementation uses two essentially independent steps: First, the determination of S_{pn} and the uncertainty ΔS_{pn} for each pixel from measured data, using the correlation between A_p and A_n . Second, the calculation of a set of calibration coefficients $\{s_p, s_n\}$ based on the set of $\{S_{pn}, \Delta S_{pn}\}$ from the first step. The former is done using a Bayesian [67] approach, and the latter is done by using a nonlinear least squares fit algorithm.

Determination of S_{pn}

For all pixels, the slope coefficient $S_{pn} = A_p/A_n$ is calculated from the data obtained from M selected events in the detector, each event consisting of two amplitude measurements $A_{p,i}, A_{n,i}$ that fulfill (4.15). One could determine S_{pn} by a straight line fit to the 2d-distribution of all amplitude pairs using a χ^2 minimization procedure. The slope of that line would correspond to S_{pn} . In practice, the resulting S_{pn} has a systematic error, because the χ^2 minimization procedure assumes a Gaussian distribution of the individual points around the fitted function, and the result can be significantly changed if some points violate this assumption. There are several suggestions in [67] to overcome this problem. In the presented implementation the probability distribution for the quantity of interest S_{pn} was calculated directly by repeated application of Bayes' theorem (4.20). This is simple to implement, because all that is needed, is a point-by-point multiplication of two functions for each event, namely the multiplication of the probability distribution for S_{pn} based on all previous events with the likelihood function of the current event. After multiplication, a normalization step follows to create a valid probability distribution that can be used in the next event.

The posterior probability distribution for S_{pn} is written as $\text{prob}(S_{pn} | \{A_p, A_n\}_M)$. Such a distribution exists for all pixels of the detector. The notation was adopted from [67], where $\text{prob}(x|y)dx$ is a function of the variable x that gives the probability of having x in the interval $[x, x + dx]$, given the condition or information y . The subset $\{A_p, A_n\}_i := \{A_{p,1}, A_{n,1}, A_{p,2}, A_{n,2}, \dots, A_{p,i}, A_{n,i}\}$ is defined as the data obtained from the first i measured events. Obviously, $\{A_p, A_n\}_M$ is the set of all measured events. The most likely value for the slope parameter S_{pn} and its error ΔS_{pn} for each p and n can be obtained from the mean and variance of the final posterior distribution. As mentioned above, the computation of the final distribution $\text{prob}(S_{pn} | \{A_p, A_n\}_M)$ is done iteratively by multiplying it with the likelihood function of each event. After each multiplication, the width of the distribution becomes narrower and the uncertainty of S_{pn} shrinks. The iteration starts with an initial guess for the $\text{prob}_0(S_{pn})$ distribution, in this case uniform within reasonable limits S_{\min} and S_{\max} . These limits should cover all occurring values of S_{pn} , which can be guessed from the maximum expected ratio of gains in all channels: $S_{\min} < \min(s_n/s_p)$ and $S_{\max} > \max(s_n/s_p)$. For example, if the smallest gain is expected to be no less than 10 times the largest gain, S_{pn} can have any value in the interval $[0.1, 10]$. The multiplication of $\text{prob}_0(S_{pn})$ with the likelihood function is the direct application of Bayes' theorem [67, 68, 69]

$$\text{prob}_i(S_{pn} | \{A_p, A_n\}_i) = \frac{\text{prob}_i(S_{pn}) L(A_{p,i}, A_{n,i} | S_{pn})}{\text{prob}(A_{p,i}, A_{n,i})}, \quad (4.20)$$

with the commonly used terminology [67]: $\text{prob}_i(S_{pn} | \{A_p, A_n\}_i)$ is called *posterior* distribution, $\text{prob}_i(S_{pn})$ is the *prior* distribution, $L(A_{p,i}, A_{n,i} | S_{pn})$ the *likelihood* function and $\text{prob}(A_{p,i}, A_{n,i})$ is a normalization factor that is also called *evidence* of the measured data. Index i indicates the iteration step. After each multiplication, the normalized posterior distribution becomes the prior for the next data point

$$\text{prob}_{i+1}(S_{pn}) = \text{prob}_i(S_{pn} | \{A_p, A_n\}_i), \quad (4.21)$$

and the final distribution is obtained at the last iteration. The likelihood function is chosen to be a Cauchy-Lorentz distribution with width w

$$L(A_{p,i}, A_{n,i} | S_{pn}) \propto \frac{1}{w^2 + \left(\log \frac{A_{p,i}}{A_{n,i}} - \log S_{pn} \right)^2}. \quad (4.22)$$

This particular choice for the likelihood function was inspired by the treatment of data with outliers in [67].

The posterior distribution has to be approximated by K points between S_{min} and S_{max} and stored during the complete procedure. This has to be done because there is no analytic expression for the intermediate shape of the distribution. Finally, according to the central limit theorem, it will approach a Gaussian distribution. The number of points K depends on the chosen limits $S_{min,max}$ and the desired accuracy of the final result. Values are in the order of a few thousand work good in practice.

Computing a set of calibration coefficients

The minimization of (4.19) is done with the nonlinear least squares fit provided by the GNU Scientific Library [34]. The set of fit parameters is $\{s_p, s_n\}$, and the input data is the complete set of measured parameters $\{S_{pn}\}$ and the uncertainties $\{\Delta S_{pn}\}$. One out of the $N_p + N_n$ parameters has to be fixed in order to find a unique solution. This was done by fixing the last p-side slopes to 1. The algorithm attempts to minimize equation (4.19) with respect to the remaining $N_p + N_n - 1$ parameters. After convergence is reached, the resulting parameter set describes the best intrinsic calibration coefficients for the individual strips relative to each other on a common arbitrary scale. The scale is determined by the channel of which the slope was set to 1. The algorithm works automatically in the following sense: Once a good parameter set (K, S_{min}, S_{max}, w) is found for a detector, intrinsic calibration coefficients can be found for it by analyzing any measured data set. The procedure does not require human assistance or supervision. The consistency of the outcome could be checked by looking at a histogram like the one shown in 4.14 b). Failures in the procedure could also be detected by checking that no extreme values of calibration parameters are obtained, i.e. values that are too close to S_{min} or S_{max} .

Offset determination

The method can be extended to take offsets in the calibration function into account. An offset o is added to equation (4.14)

$$E = o + sA. \quad (4.23)$$

If offsets are allowed in the energy dependence of the amplitude, the linear dependence between A_p and A_n is also allowed to have an offset O_{pn} . This changes equation (4.17) into (see figure 4.13 b)

$$A_p = O_{pn} + S_{pn} A_n. \quad (4.24)$$

The determination of O_{pn} and S_{pn} , can be achieved with two-dimensional probability density functions $p(S_{pn}, O_{pn} | \{A_p, A_n\})$. The basic procedure is the same as described in 4.1.4, only that

the number of supporting points of the distribution is larger, as it is two-dimensional. For the same reason, the calculation is much slower, because the point-wise multiplication takes more time. The likelihood function has to be extended from (4.22) to be two-dimensional as well, taking into account the correlation between offset and slope:

$$L(A_{p,i}, A_{n,i} | S_{pn}, O_{pn}) \propto \frac{1}{w^2 + \left(\log \frac{A_{p,i} - O_{pn}}{A_{n,i}} - \log S_{pn} \right)^2}. \quad (4.25)$$

For the numerical approximation, a range of possible offset parameters O_{pn} has to be specified by two additional parameters, O_{min} and O_{max} . These limits have to be chosen such, that they include all occurring O_{pn} as defined in (4.26). Even if the numerical approximation of the two-dimensional posterior distribution requires significantly more memory and computation time than in the one-dimensional case, it is possible on nowadays desktop computers.

The calculation of the calibration coefficients for each strip has to be modified as well. All relevant quantities are related as in equation (4.18), but with an additional equation for the offset:

$$S_{pn} = \frac{s_n}{s_p} \text{ and } O_{pn} = \frac{o_n - o_p}{s_p}. \quad (4.26)$$

With the modified relations of equation (4.26), the set of $2N_p + 2N_n - 2$ calibration parameters $\{o_p, s_p, o_n, s_n\}$ can be found by minimizing equation (4.27), which is an extended form of equation (4.19):

$$\chi^2 = \sum_{p,n} \left(\frac{S_{pn} - \frac{s_n}{s_p}}{\Delta S_{pn}} \right)^2 + \left(\frac{O_{pn} - \frac{o_n - o_p}{s_p}}{\Delta O_{pn}} \right)^2. \quad (4.27)$$

where S_{pn}, O_{pn} and $\Delta S_{pn}, \Delta O_{pn}$ are mean and variance of the final distribution $p(S_{pn}, O_{pn} | \{A_p, A_n\})$. This time, two parameters have to be fixed to find a unique solution, for example the offset and slope of the last strip on the n-side $o_N = 0$ and $s_N = 1$.

The effect of the intrinsic selfcalibration can be seen for the Target-DSSSD detector in figure 4.14.

Possible Improvements

The procedure described here, requires the user to find good parameters for the range of the probability distributions and their density of points. If the range is not known, the user has to choose a wide and fine enough grid for the representation of the distribution and pays with longer computation time and more memory consumption. Therefore, it would be a significant improvement in performance and usability, if the probability distributions would be adaptive in range and density of points. With such an improvement implementation, the only remaining parameter would be the width w of the likelihood function in Eqs. (4.22,4.25).

4.1.5 Absolute calibration of the LYCCA DSSSD detectors

The procedure as described in this section was applied to all of the individual wall DSSSD detectors. After that, the spectra of all 16 detectors were scaled by hand to achieve a DSSSD

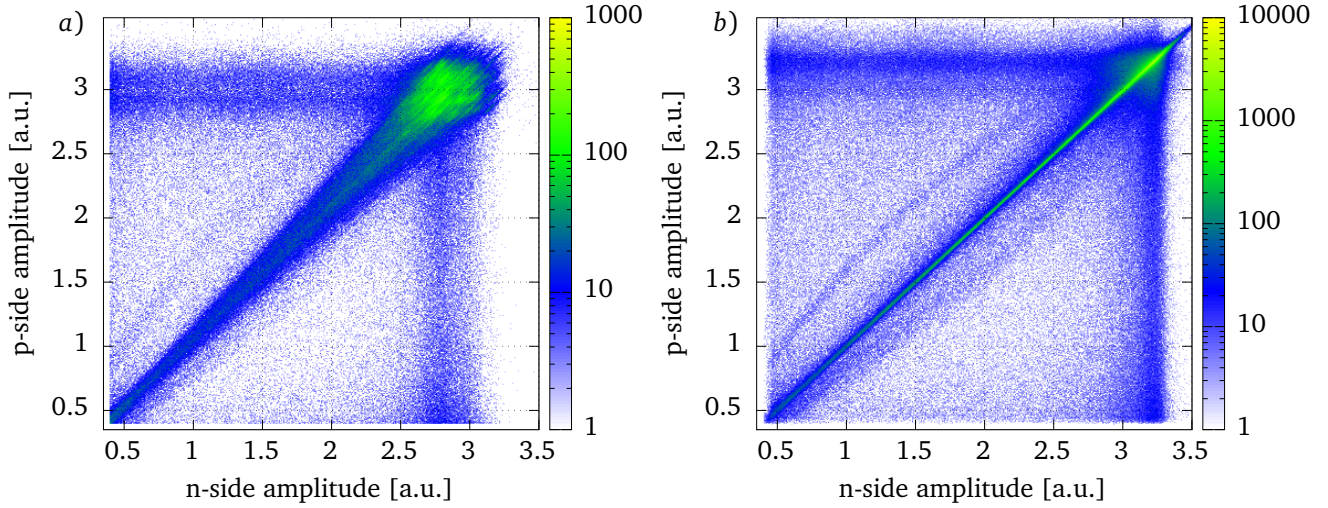


Figure 4.14.: Comparison of the p-side and n-side amplitudes of the target-DSSSD detector of LYCCA, on the left hand side without calibration coefficients applied, and on the right hand side using the coefficients obtained by the intrinsic selfcalibration method. The improvement is obvious: In the uncalibrated detector, each pixel has a different slope of the n-p-side correlation. After calibrations, all slopes are aligned.

gain matching. A final scaling factor was applied, such that the ^{80}Kr component of the fragments was at the same energy as the ATIMA prediction as shown in figure 4.3 for the measured velocity.

Note that the amount of work for absolute calibration was reduced by a factor of 32, because of the intrinsic calibration. Furthermore, no special calibration runs are needed to apply this procedure. Consequently applying this technique leaves more valuable beam time for physics data taking.

4.1.6 CsI calibration

In a first step, all 144 CsI spectra were gain matched by hand, such that the peak of the ^{80}Kr component of the fragments were at the same position in all of their raw spectra. Then, a scaling factor was applied to match the kinetic energy as calculated from the measured β for ^{80}Kr . CsI scintillators have a non-linear light yield when exposed to heavy-ion radiation. If the CsI signal should match the kinetic energy of the projectile regardless of the isotope, the energy dependent light yield has to be calibrated. The Z and energy dependence of the light yield response of CsI to heavy ions was measured for example in [59] and [60] for energies up to 1250 MeV and $Z \leq 54$. For higher energies it is claimed that the light yield behaves linear. During the calibration of LYCCA, it was observed that this is not so for heavy ions in typical PreSPEC experiments. An important part of the absolute calibration of the CsI detectors was thus the determination of the non-linearity and Z dependence of the CsI light yield for heavy ions up to 15 GeV. LYCCA allows to calibrate its own light yield curve only using its own data during a production run. Qualitative particle identification can be achieved without absolute calibration of all detector systems by gating on the structures in the correlation spectra. These can be used to identify isotopes for each event. Using a calibrated velocity measurement, together with the isotope gates, allows the calculation of the kinetic energy of the heavy ion in

each event. By correlating the kinetic energy with the CsI amplitude which is proportional to the light yield, one can measure the light yield curve for all detected elements. Since the energy of the recoils after the reaction in the secondary target is widely distributed, one can obtain light yield curves for all elements over a large range of kinetic energy. This data can be used to fit an empirical parameterization of the dependence of the kinetic energy $E_{\text{kin}}(L_{\text{CsI}}, Z)$ on the light yield L_{CsI} and the element Z . Such a function can be used to get an absolute calibration of the CsI detectors that is needed for particle identification.

The data and the fitted parameterization are shown in figure 4.15. For observed range of ion energies, the CsI light yield is not linear anymore. In the phenomenological parameterization it was assumed that the data behaves linearly with one slope below a certain threshold, and linearly with another slope above that threshold. Both regions are smoothly connected with a smoothness parameter σ . The fitted parameterization uses this two-slope model for one value of Z . Z dependence is introduced by scaling the resulting curve with Z^κ , which introduces another parameter.

$$E_{\text{CsI}} = \Lambda_\sigma (\alpha L_{\text{CsI}}, \zeta + \xi L_{\text{CsI}}) Z^\kappa \quad (4.28)$$

$$\text{with } \Lambda_\sigma(f, g) = -\sigma \ln(e^{-f/\sigma} + e^{-g/\sigma}), \quad (4.29)$$

where the function Λ_σ is the “smooth minimum” function. The fitting procedure to all data results in the following values for the parameters:

$$\alpha = 0.1473(3) \text{ MeV} \quad (4.30)$$

$$\zeta = 503(4) \text{ MeV}$$

$$\xi = 0.0849(4) \text{ MeV}$$

$$\sigma = 104(3) \text{ MeV}$$

$$\kappa = 0.5788(4)$$

The Z dependence can be explained, at least partially, by the energy loss in the LYCCA ToF-stop and DSSSD detectors that are directly in front of the CsI crystals. Higher values of Z have a larger stopping power ($\propto Z^2$), and the remaining kinetic energy in the CsI detectors is smaller. A comparison with the parameterization of the CsI light yield of ref. [60] shows qualitative agreement with the measured LYCCA data. The measurements in that reference were done with energies up to 50 MeV/u, and the parameterization differs mainly in the smaller Z dependence compared to the LYCCA data, which indicates thinner dead-layer in their setup. The curvature at the bottom end of the energy spectrum cannot be observed with LYCCA because of the different energy range. The only major difference is the non-linearity for higher energies in the LYCCA setup, and it is not clear if that is an effect of the CsI material or of the PIN diodes that were used for the readout. Regardless of the exact reasons, a correction of the CsI signals for non-linearity and Z dependence is needed in order to get a global mass calibration as shown in figure 4.4 b).

4.1.7 Bayesian LYCCA isotope identification

This section describes how the correlation between quantities measured by LYCCA can be exploited to decrease the uncertainty of particle identification. This is of great interest as it would increase the range of possible experiments towards higher masses and charges, at no additional cost in instrumentation.

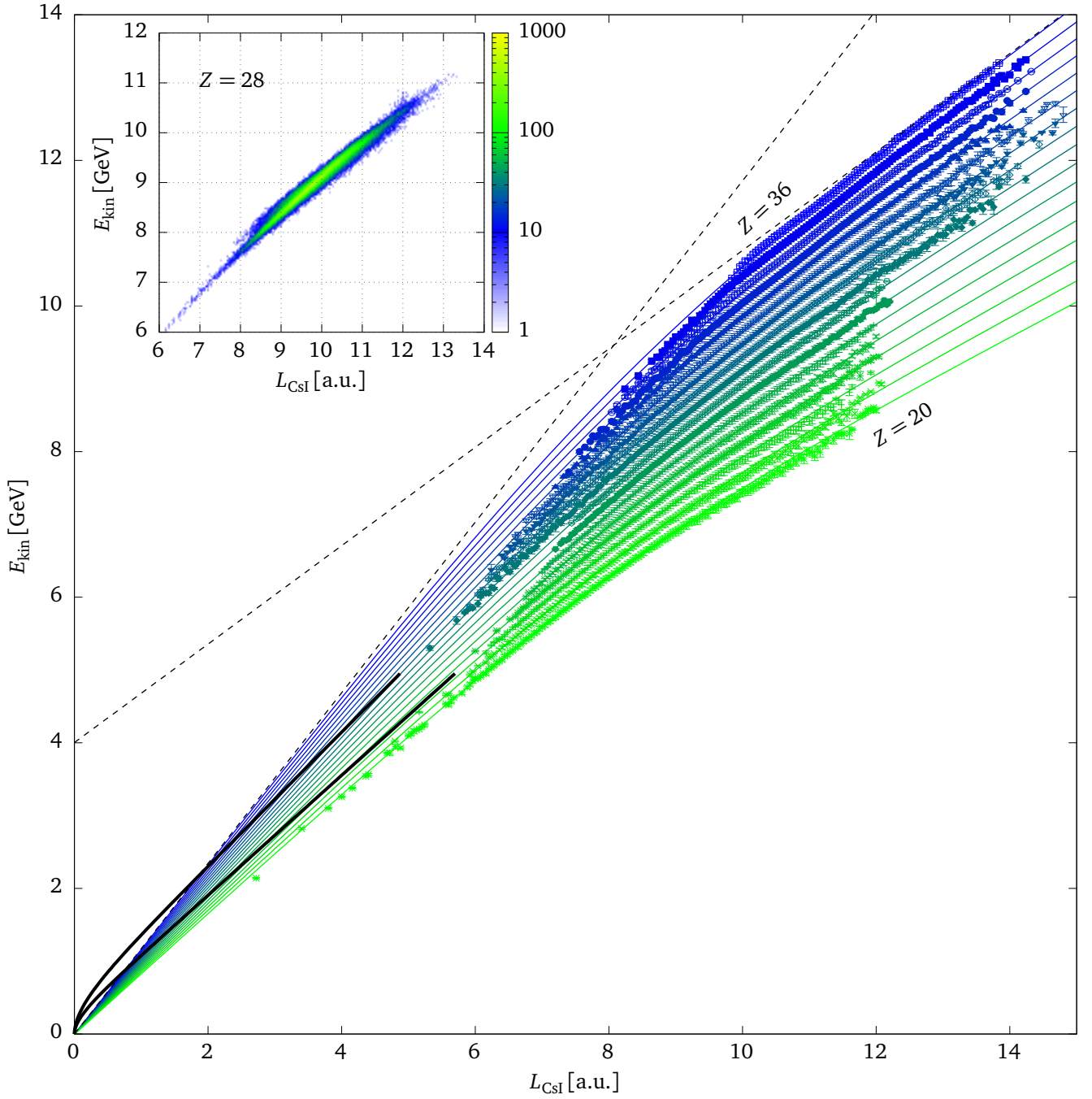


Figure 4.15.: Kinetic energy E_{kin} of ions in correlation to their observed light yield L_{CsI} in the CsI detectors. Data points were extracted from Z -gated $E_{\text{kin}}-L_{\text{CsI}}$ histograms as shown in the inset for $Z = 28$. Error bars are scaled down by a factor two for better visibility. Information about the kinetic energy is obtained from A and β using equation (4.6) minus the measured energy loss in the DSSSD. Black, dashed lines show the asymptotic behavior for $Z = 36$. The thick solid, black lines are drawn using the parameterization of the CsI light yield from ref. [60], which was fitted to data at lower energies. The upper curve is for $Z = 36$, and the lower curve for $Z = 20$. Note that it is difficult to compare both parameterizations, because the light yield may be scaled by an arbitrary factor. However, it is obvious that the Z dependence is different.

The classification of events that are represented by multidimensional vectors is known as multivariate analysis. In this case the event vector consists of the measured quantities by LYCCA $(\beta, \Delta E, L_{\text{CsI}})$, and the different classes are two positive integer values for the mass and charge: (A, Z) . In this section, a classifier for LYCCA data is developed based on the Bayes' theorem. It calculates for each event the probability to belong to class (A, Z) . The event is then assigned to the class with maximum probability $(A_{\text{max}}, Z_{\text{max}})$. This method is known in the literature as projective likelihood estimator, or classifier, or also as naive Bayes estimator [70].

Almost all research about the general problem of multivariate analysis and the optimization of classifiers is done in particle physics. For some (unknown) reason, it is uncommon to use such techniques in the field of nuclear structure research. In nuclear structure research, one uses mainly the “polygonal cut” classifier, which is the simplest of all thinkable classifiers. In view of the fact that the LYCCA device was used up to, and even beyond its specified limit for mass and charge resolution, in particular in the experiment S429 (see table 4.1), and in view of the overall expenses for all these experiments, it is well justified to spend some time in developing more advanced techniques for event classification.

The underlying idea is to use the available information $(\beta, \Delta E, L_{\text{CsI}})$ in the best possible way in order to infer the quantity of interest $(A_{\text{max}}, Z_{\text{max}})$. This reduces the uncertainty in particle assignment, because it makes use of the physical correlation between the measured quantities. This can be seen as a generalization of the fact that repeated, independent measurements of any quantity reduce the uncertainty in the final best estimate for that quantity.

As shown in the previous sections, analytical expressions for the relations between all of the quantities measured by LYCCA are known. These relations allow the direct computation of the probability to obtain the measured vector $(\beta, \Delta E, L_{\text{CsI}})$ in an event, assuming that the ion had a given velocity, charge and mass: (β_0, A, Z) . Doing this for all possible values of (β_0, A, Z) inside a reasonable range around the conventionally determined¹ values allows to determine the most likely ion species, as well as it's most likely velocity β_0 .

Formalism

The formalism is using conditional probability distributions as already introduced in section 4.1.4. In the following, $(\beta, \Delta E, L_{\text{CsI}})$ denotes the vector of measured data by LYCCA: velocity, energy loss and CsI light-yield, respectively. Given this data vector, probability distributions can be calculated for the true value of the quantities: (β_0, A, Z) . In order to illustrate this, consider the simplified example of a reduced data vector $(\beta, \Delta E)$, and a classification limited to the integer value of Z . The relation between these quantities $\Delta E(\beta, Z)$ is known and given by equation (4.4). The likelihood for having measured $(\beta, \Delta E)$, given a true velocity and charge number (β_0, Z) , is given by the two dimensional, uncorrelated Gaussian distribution

$$\text{prob}(\beta, \Delta E | \beta_0, Z) = \frac{1}{2\pi \sigma_\beta \sigma_{\Delta E}} \exp \left[-\frac{1}{2} \left(\frac{\beta - \beta_0}{\sigma_\beta} \right)^2 - \frac{1}{2} \left(\frac{\Delta E_0 - \chi \frac{Z^2}{\beta^\kappa}}{\sigma_{\Delta E}} \right)^2 \right]. \quad (4.31)$$

There is no correlation, because β and ΔE are measurements from two independent detector systems. The most likely value for Z can be found by searching for the global maximum of equation (4.31). This is visualized in figure 4.16. Equation (4.31) can be used as likelihood

¹ In principle, call combinations of (β_0, A, Z) should be considered, however, in practice it is sufficient to only compute the values around the conventionally determined values.

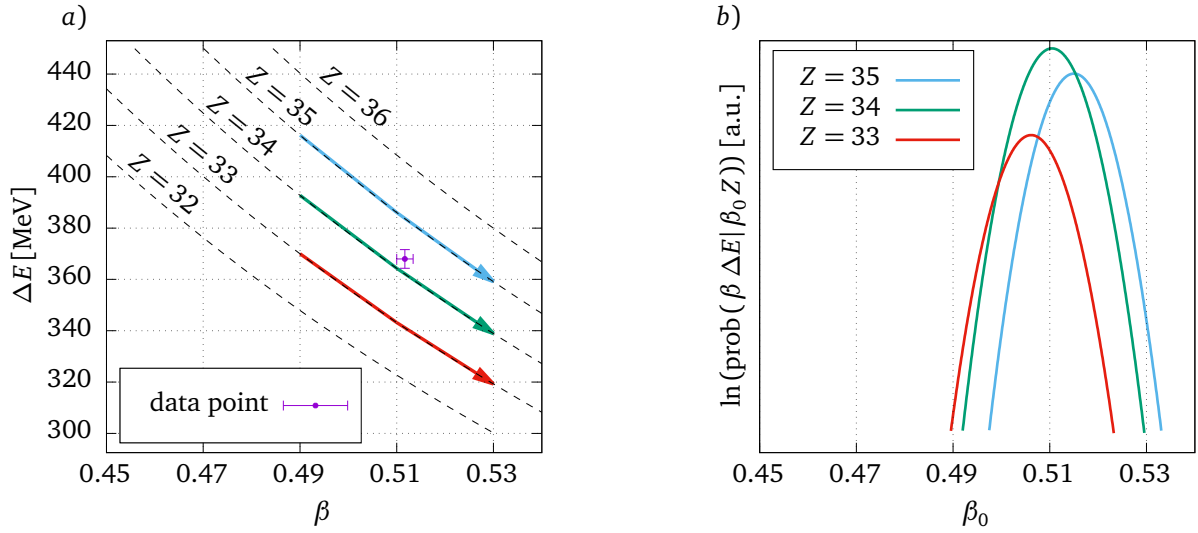


Figure 4.16.: Visualization of the described formalism to gain information from two independent measurements of correlated quantities. In panel a), an event was detected (magenta point) with known uncertainties. Combinations of physically possible values are drawn as dashed lines. These combinations are parameterized by the nuclear charge Z and the “true” velocity β_0 . Panel b) shows the result of evaluating equation (4.31) for three different values of Z and β_0 . From the curves it is clear that the highest absolute probability is reached for $Z = 34$. It can also be seen that the maximum of the probability curve for $Z = 34$ is slightly shifted to the left, i.e. taking all information into account, best estimate of the particle velocity is also improved.

function in Bayes’ theorem to calculate the posterior probability distribution for the particles velocity and charge number (β_0, Z).

In order to extend the formalism to all three quantities, equations (4.6) and (4.28) have to be included and the likelihood function is then:

$$\text{prob}(\beta, \Delta E, E_{\text{Csl}} | \beta_0, A, Z) = \frac{1}{(2\pi)^{3/2} \sigma_\beta \sigma_{\Delta E} \sigma_{L_{\text{Csl}}}} \quad (4.32)$$

$$\times \exp \left[-\frac{1}{2} \left(\frac{\beta - \beta_0}{\sigma_\beta} \right)^2 - \frac{1}{2} \left(\frac{\Delta E - \chi \frac{Z^2}{\beta_0^{\kappa}}}{\sigma_{\Delta E}} \right)^2 - \frac{1}{2} \left(\frac{(\gamma - 1) A u c^2 - \overbrace{(E_{\text{Csl}}(L_{\text{Csl}}) + \Delta E)}^{=E_{\text{kin}}}}{\sigma_{L_{\text{Csl}}}} \right)^2 \right],$$

where $E_{\text{Csl}}(L_{\text{Csl}})$ is given by equation (4.28). Equation (4.32) is the likelihood function for the measurement of a certain LYCCA event. Using Bayes’ theorem, the probability distribution for (β_0, A, Z) can be obtained:

$$\text{prob}(\beta_0, A, Z | \beta, \Delta E, E_{\text{Csl}}) = \frac{\text{prob}(\beta, \Delta E, E_{\text{Csl}} | \beta_0, A, Z) \text{prob}(\beta_0, A, Z)}{\text{prob}(\beta, \Delta E, E_{\text{Csl}})} \quad (4.33)$$

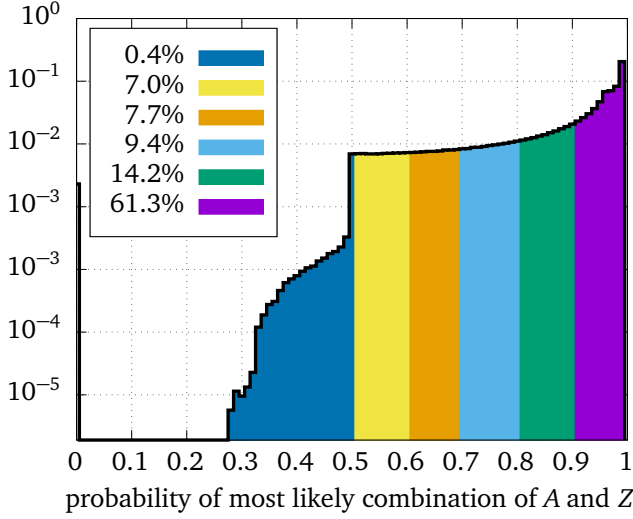


Figure 4.17.: Histogram of the probability of the most likely assignment for A and Z using the presented method. The fraction of events in each colored area is given. It can be seen that 61.3% of the events can have a combination of A and Z assigned, which has probability of more than 90% to be true. Already 75.5% of the A and Z assignments have a probability of more than 80% (the sum of the magenta and green area).

which is proportional to the likelihood function (4.32) if a uniform prior distribution is assumed. The maximum of (4.32) can be determined numerically for every event. Because equation (4.33) is a probability distribution, it has to be normalized to one:

$$\int_{\beta_0} d\beta_0 \sum_A \sum_Z \text{prob}(\beta_0, A, Z | \beta, \Delta E, E_{\text{CSL}}) = 1. \quad (4.34)$$

One can use (4.34) in order to calculate the evidence, i.e. the normalization factor, and get a quantitative value of the probability for each combination (A, Z) :

$$\text{prob}(\beta, \Delta E, E_{\text{kin}}) = \int_{\beta_0} d\beta_0 \sum_A \sum_Z \text{prob}(\beta, \Delta E, E_{\text{CSL}} | \beta_0, A, Z). \quad (4.35)$$

This allows other parts of the analysis to specify an isotope condition in terms of a required probability for a certain isotope, such as $\text{prob}(A = 74, Z = 34 | \beta, \Delta E, E_{\text{CSL}}) > x$, where x is a percentage, and the probability can be computed by marginalizing the β_0 dependence:

$$\text{prob}(A, Z | \beta, \Delta E, E_{\text{CSL}}) = \int_{\beta_0} d\beta_0 \text{prob}(\beta_0, A, Z | \beta, \Delta E, E_{\text{CSL}}). \quad (4.36)$$

The integral and the summation in (4.35) can be calculated numerically for each event. Figure 4.17 shows the probability distribution for the most likely combination of A and Z as obtained with the described method for all events.

Figure 4.18 emphasizes the difference between the Bayesian classifier and the “graphical cut” classifier. Here, the conventional particle identification plot is shown without any gate, and with two different conditions on the particle assignment probability. It can be seen, that the selection of events is not always sharp in the (A, Z) representation.

Figures 4.19 and 4.20 show, how the total charge and mass histograms can be separated in individual histograms that are associated to a single value of A or Z . This shows again, that the distributions of most likely assignments are overlapping.

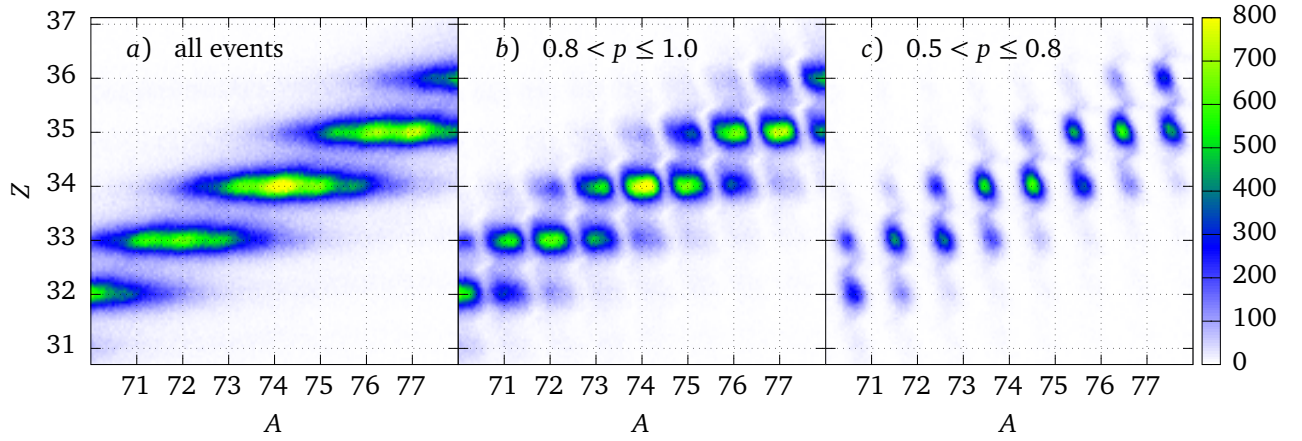


Figure 4.18.: Visualization of the effect on the conventional particle ID histogram when gated on different assignment probabilities: the histogram in panel a) shows the conventionally obtained particle ID. In panel b), the histogram contains only events where the assignment probability $p := \text{prob}(A_{\text{max}}, Z_{\text{max}} | \beta, \Delta E, E_{\text{CSl}})$ was at least 80%. In panel c) only events with p less than 80% are shown. These three histograms point out the difference between a classifier defined by a graphical cut in the A vs. Z histogram and the Bayesian classifier: in the latter case, as can be seen in b) and c) there is no sharp cut between the different peaks. Note that event if the peaks in panel c) are in the middle between two integer numbers, the events therein are still assigned to a mass and charge number with a probability of more than 50%, according to figure 4.17.

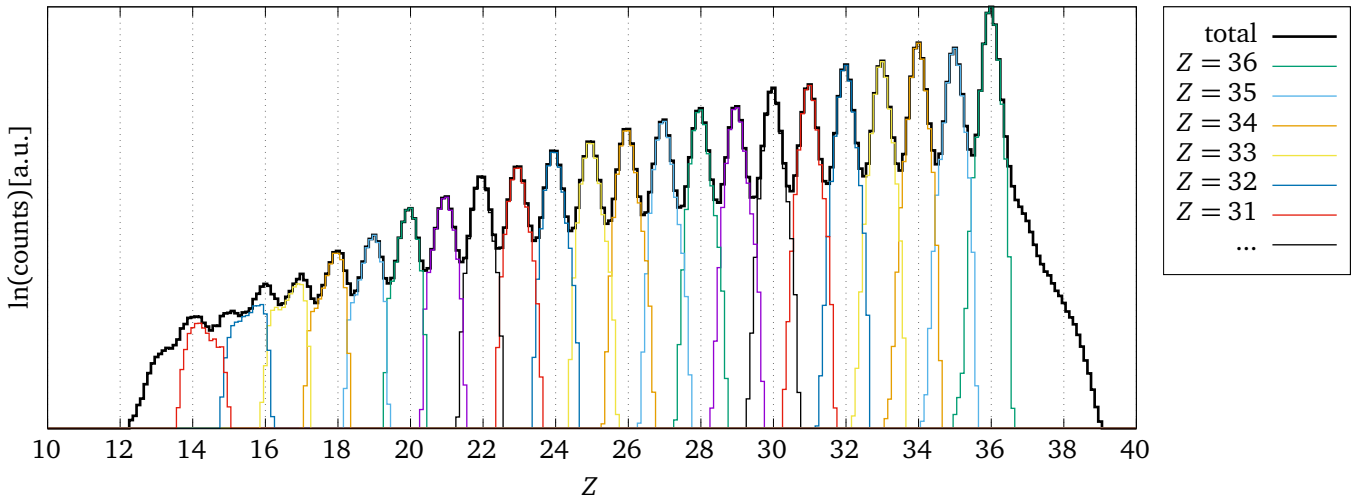


Figure 4.19.: The total nuclear charge distribution, as determined by equation (4.5) of particles after the secondary reaction target with a particle- γ trigger is shown by the black histogram. The colored histograms show the charge distribution, determined by (4.5), if the likelihood was largest for a fixed value of Z . Here, the advantage might not be obvious, but in the case of a more severe overlap of the peaks, as is the case for the mass distribution in figure 4.20, the events can be assigned to the most likely value, even if they are closer to the center of a neighboring peak.

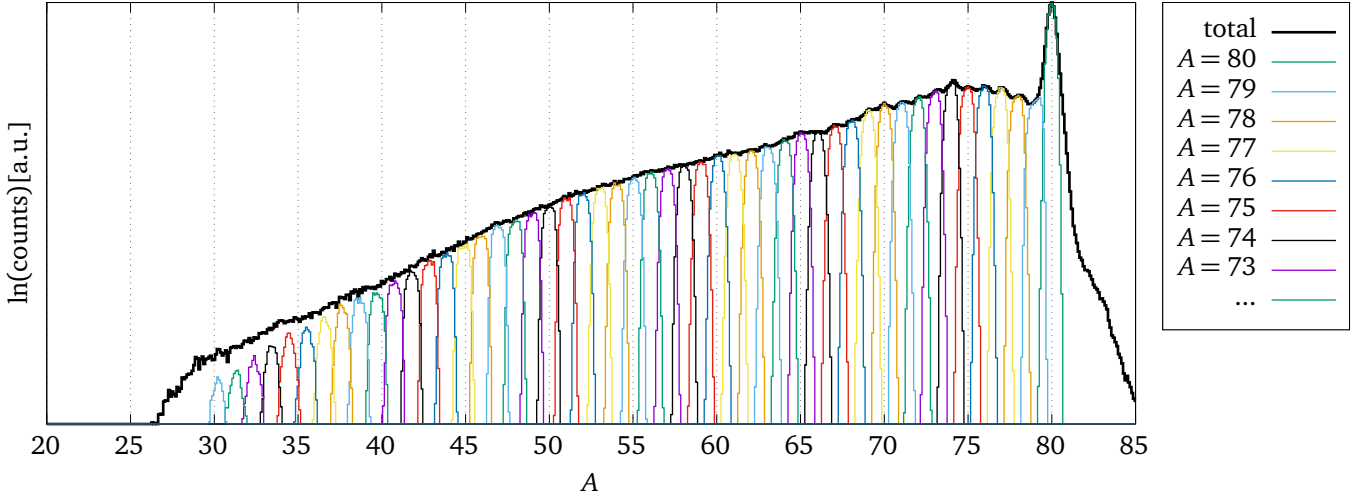


Figure 4.20.: The total nuclear mass distribution, as determined by equation (4.6) of particles after the secondary reaction target with a particle- γ trigger is shown by the black histogram. The colored histograms show the mass distribution, determined by (4.6), if the likelihood was largest for a fixed value of A .

Discussion

It is difficult to make a direct comparison between the conventional and the Bayesian method to determine A and Z with LYCCA. However, there are some obvious advantages of the Bayesian method: First, in principle, the value for β_0 at which $\text{prob}(\beta_0, A, Z | \beta, \Delta E, E_{\text{CsI}})$ has its maximum, should be a better estimate for the particle velocity than the value of β as it was measured by the time-of-flight detectors because correlated information from the DSSSD and CsI detectors are folded into the final value of β_0 . More additional information comes from the fact that A and Z have to be integer values, constraining the space of possible combinations of the measured quantities. The improved value of the particle velocity can help to make a more precise Doppler correction of the measured γ -rays. Second, as shown in figures 4.18, 4.19 and 4.20, the Bayesian classifier allows for a more precise, more quantitative and more efficient particle selection. The precision and quantification comes from the fact that the gate is not a hand-drawn line around a diffuse spot in a histogram, but a mathematical statement of how likely an A and Z assignment should be. The efficiency should be increased because of the increased resolving power of LYCCA if all available information, including the correlations between all measured quantities, are used. This becomes even more important in the case of experiments with heavier ions, where the limited resolution of LYCCA does not allow to see any structure in the particle identification plots. Even without visible structures, a Bayesian classifier allows to make quantitative decisions if an event is to be treated as background or signal event.

The downside of the quantitative estimate of probabilities is, that they are only correct if all included quantities are correct. This is particularly true for the uncertainties in the measurements: $\sigma_\beta \sigma_{\Delta E} \sigma_{L_{\text{CsI}}}$. The quantitative value of the probability of the assignment of $(A_{\text{max}}, Z_{\text{max}})$ depends very strongly on these uncertainties. Further, the likelihood function (4.32) assumes that all uncertainties are Gaussian with known width. This is not necessarily the case. However, even if the interpretation as probability is not quantitatively correct, the calculated value can still be used as an event classifier.

4.2 The Advanced Gamma Tracking Array (AGATA)

From 2012 on, the AGATA detector is used as the main γ -ray spectrometer in the PreSPEC setup. Only few details about this detector will be given here. More information is available in [71] and [22].

AGATA is a high-purity germanium (HPGe) γ -ray detector array, which is developed, build and maintained by an European collaboration of nuclear physics research institutes, represented by the AGATA Collaboration Council (ACC) (for more information, visit the ACC website [72]). The detector system is designed to overcome known deficiencies of the previous generation of Compton-suppressed HPGe arrays, where each HPGe crystal of the array is surrounded by scintillation material that acts as active anti Compton shield, as shown in figure 4.21 b). A signal in the surrounding scintillator in coincidence with a signal in the surrounded HPGe detector will identify events, in which only a fraction of the photon energy was deposited inside the germanium. These events would contribute to the background in the spectrum and can be rejected. The shielding scintillators take a considerable fraction of the total solid angle and limit the total peak efficiency of the detector array to no more than 10% [73].

In AGATA, pulse-shape analysis (PSA) and γ -ray tracking are applied to overcome these limitations. With PSA, the locations of the γ -ray interactions inside the germanium crystals can be determined, and a γ -ray tracking algorithm reconstructs the most likely time ordering of these interactions to obtain the track of the photon inside the detector. A tracking setup is schematically shown in figure 4.21 c). PSA techniques require segmented detectors, because the position information is only obtained by analyzing the shape of several segment signals. In a detector

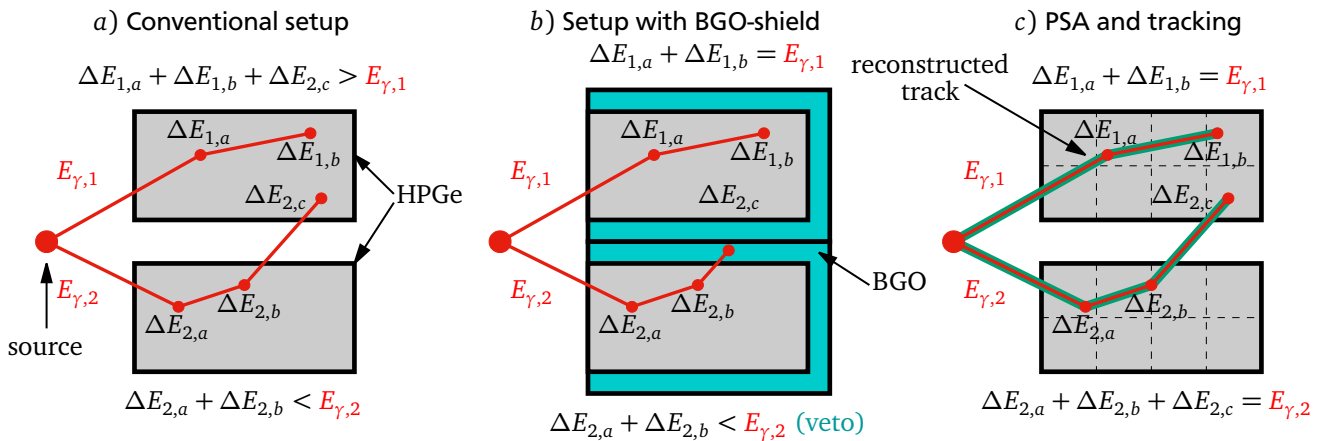
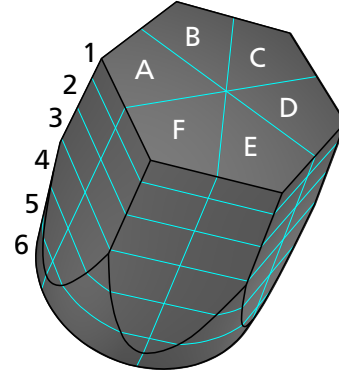


Figure 4.21.: Schematic explanation of the benefits of pulse-shape analysis and γ -ray tracking over the use of BGO anti-Compton shields. The example shows the response of two HPGe detectors on two photons if one of the photons cross-scatters into another detector: a) non-segmented, non-shielded detectors, where both detectors detect a wrong amplitude in the shown case. b) non-segmented, Compton shielded detectors, where one detector measures the correct amplitude, and the other detector reading is suppressed. c) Segmented tracking detector array, where the interaction points inside the crystals are reconstructed with pulse-shape analysis (PSA) and the most likely photon paths are calculated. The energy of the incoming photons is calculated as the sum of the energy deposition at the interaction points along each track.

Figure 4.22.: Geometry of one AGATA crystal and the contact segmentation. Blue lines indicate segment boundaries. The segment labeling uses numbers 1-6 for the different rings, and letters from A to F for the sections in each ring.



array with PSA and γ -ray tracking, there is no need for active anti-Compton shields, because scattering from one detector to the other will be considered in the track reconstruction, and is not different from the track reconstruction within one crystal. This allows to fill almost the full solid angle of 4π with high resolution germanium detector material, which is expected to result in a total efficiency of about 50%. An additional advantage for the present case of in-beam γ -ray spectroscopy is the high angular resolution in the detection of the γ emission angle because of the ability to determine the first interaction point in the crystal with high accuracy. This reduces one source of uncertainty in the Doppler correction of the detected γ -rays. In the RISING HPGe array, which did not have PSA, this uncertainty was reduced by increasing the distance of the detectors to the target at the cost of reduced detection efficiency.

AGATA HPGe crystals are made of n-type HPGe material, with closed-ended coaxial shape, segmented outer p^+ contacts with 6 rings and 6 segments (see fig. 4.22), and an inner n^+ core contact. The position sensitivity is achieved by analyzing the rising edge of the pre-amplifier signals from all 37 contacts, which depends on the charge carrier movement inside the crystal volume, and thereby carries information about the position of charge carrier creation by the detected photon. All pre-amplifier signals are recorded with flash ADCs at a sampling rate of 100 MHz and 14-bit resolution. Position information is obtained by the method of pulse-shape comparison, where the 100 first samples of all 37 pre-amplifier signals are compared against a data base with simulated signal shapes corresponding to different interaction points. That database covers the complete crystal volume with a grid size of 2 mm in all 3 spatial dimensions.

4.2.1 Calibration

AGATA requires in addition to the gain calibration for each segment and each core, a correction of the cross-talk between all electrical signals of each crystal, a time alignment of the digitized pre-amplifier signals and a correction for charge carrier trapping in crystal defects caused by fast neutrons. These corrections depend on adjustable parameters that have to be optimized. All relevant data to do these corrections, including the digitized signal waveforms, are recorded and allow a so called replay of the data with different parameters. This allows to optimize the performance of the AGATA spectrometer after the experiment. Improvements in energy and position resolution can be expected after optimizing the parameters of PSA, which should be done in order to achieve the best possible result. Details about the steps involved in AGATA data replay can be found in [74] and the references therein.

However, this was not the focus of the present work, and all shown results will rely on the online configuration of the AGATA analysis as it was configured during the experiment.

4.2.2 Simplified γ tracking scheme

Instead of the γ -ray (MGT) tracking that calculates the most likely path of the photon inside the germanium crystal, a simplified scheme was applied, where the first interaction point inside the crystal is assumed to be the one with the highest energy deposition of all interaction points. The incident photon energy is assumed to be given by the core energy signal, i.e. it is assumed that the photon deposited its full energy inside one single crystal. The full tracking should be slightly more efficient (see [75]) than the simplified scheme because of the cross-crystal Compton scattering. Because of the low expected benefit, the simplified scheme is preferred in the presented analysis.

4.3 PreSPEC data acquisition

Even though the data acquisition system (DAQ) was certainly not part of this work, some knowledge information about it is needed in order to analyze PreSPEC-AGATA data. In the following, details about the DAQ system are discussed as far as they are relevant for the data analysis. More information about the hardware and the trigger logic can be found in [76], or in identical form in chapter 3 of [53].

The Multi-Branch System (MBS) [77, 78] was used to acquire data from the FRS, LYCCA and HECTOR, while AGATA has its custom DAQ system based on NARVAL (Nouvelle Acquisition temps-Réel Version 1.2 Avec Linux) [27].

4.3.1 MBS

MBS was developed at GSI and introduced in 1994, to serve as a general purpose data acquisition system, suitable to most experiments at the facility. It is a combination of hard and software systems to read out and combine the data from several VME² crates. All VME crates in the system need to have one TRIVA module [79] and a MBS compatible crate controller, such as RIO3 or RIO4 embedded Power Architecture computer. MBS reads data from all separate VME crates and combines them into a single event. The data readout is trigger based and can distinguish up to 14 different triggers, numbered from 1 to 14. Each trigger initiates a data readout of a subset of data, and is created based on the presence of coincident signals in one or more detectors in the setup. For example, in the PreSPEC setup, a signal in scintillator S41 in coincidence with at least one signal in AGATA, created the trigger 7. The same signals, and additional signals in at least one LYCCA Wall DSSSD and a signal in the LYCCA ToF-Start detector created the trigger 9. Triggers are prioritized according to their number, i.e. if all conditions for trigger 7 and for trigger 9 are fulfilled, the event will obtain number 9. Always the highest of all possible trigger numbers will be chosen for any given event. The logic circuitry, which generated trigger numbers from coincident signals was implemented inside the FPGA of a VULOM4 VME module with the TRLO firmware [80, PHN-IS-EE-03]. During the readout process, the acquisition system

² MBS also supports different systems, such as CAMAC

is busy and cannot be triggered again for a duration that depends on the amount of data to be read out. This is the so-called dead-time period. The time between two successive events cannot be shorter than the dead-time. The fraction of time, when the system is busy is called dead-time ratio, and the fraction of time when the system is not busy is called life-time ratio. The implications of this are discussed for two different runs of the S424 experiment in section 5.6.7.

4.3.2 AGATA

Data flow

The AGATA detector comes with custom detector support system (DSS), custom front-end electronics (FEE) and custom data acquisition system (DAQ). The front-end consists of custom pre-amplifiers for the core and segment signals and custom digitizer modules. There is one digitizer module for each crystal, with 38 channels for the 36 segment signals and two core signals. The digitizers are sampling with 100 MSPS and continuously stream sampled data through a fiber-optic link to custom pre-processing cards in Advanced Telecom Computing Architecture (ATCA) crates. The pre-processing cards apply conventional digital moving window deconvolution filters to the two independent low and high gain core signals to extract energy information³. The pre-processing cards are connected to the global trigger and synchronization system (GTS), which distributes a global clock signal over a network with tree topology from the root node to all connected pre-processing cards and to the AGATA module which will be described below. Pre-processing and AGATA modules are located at the leaf positions of the tree with identical distance (number of intermediate nodes) to the root node. The global clock is synchronized to the sampling clock in the digitizers, so each recorded signal trace has a global GTS timestamp. The GTS timestamp has 48-bits and a resolution of 10 ns. It will overflow after ≈ 781 hours, i.e. a full experiment can be done with each event having an absolute and unique time stamp with 10 ns resolution. In principle, GTS can also take trigger decisions, but this possibility was not used in the PreSPEC-AGATA setup because of the high latency of the digital electronics compared to the analog trigger generation from the measured particle signals. The GTS root node was configured to validate all trigger requests generated by the AGATA module inside the MBS system and write all AGATA crystals with a core signal above a noise threshold. The MBS γ trigger was created from discriminated inspection line signals of the digitizers modules. The inspection lines duplicate the core signal at the digitizer input. This was done to get a low latency γ trigger that comes at approximately the same time as all other trigger signals from HECTOR, LYCCA and FRS. The trigger from the inspection lines is the main AGATA γ trigger.

From the preprocessing cards, the validated data is transferred, again with an optical link, to custom PCI Express interface cards in the computers of the processing farm. The computers in the farm run the NARVAL (Nouvelle Acquisition temps-Réel Version 1.2 Avec Linux) [27] system, configured to perform a certain number of task for the individual crystal data: preprocessing, pulse-shape analysis, neutron-damage correction. This is called local level processing. In other setups with AGATA, the processing continues by merging the data from individual crystals to form a complete γ event with all interaction positions in the detector array, on which γ -ray tracking is applied. The tracking algorithm needs the source position of the γ -ray, which

³ The energy information can be used to make a trigger decision, but in the PreSPEC setup the γ trigger is created in a different way

is known in other setups by the position of the beam on the target. In the PreSPEC setup, the beam spot is very large, and the position of each γ emitting projectile has to be determined by the LYCCA target DSSSD. In order to analyze DSSSD data on the NARVAL farm, the MBS data has to be streamed into the farm. This data transfer was only working online during the last beam time in 2012 (S428 and S431), after experiment S424 was already over. For that reason another approach was used: AGATA data was stored at the crystal level (one adf file per crystal), and the software for offline analysis was opening all AGATA data files and the MBS data file at the same time, merging the events based on the GTS timestamp and analyzing the combined γ -particle data. In this approach the γ -ray tracking algorithm had to be integrated in the offline analysis software.

Data format

offset (bytes)	size (bytes)	C type	meaning
0	4	<code>uint32_t</code>	length of the frame in 32-bit words (header included)
4	4	<code>uint32_t</code>	specifies the frame content (0xfa010102 for PSA frame)
8	4	<code>uint32_t</code>	event number
12	8	<code>uint64_t</code>	GTS timestamp

Table 4.3.: Content of the adf frame header.

From the point of data analysis, the most important information is the format of the data written on disk during the experiment. For AGATA, the data format is in principle documented inside the GammaWare user manual [81, Appendix B]. But that reference does not completely match the data on disk from the S424 experiment. Therefore, the format of AGATA data at the crystal level, i.e. what NARVAL writes after local level processing, is fully documented here.

AGATA adf files are a series of frames, with a common frame header and different frame contents. The frame header always contains four numbers as shown in table 4.3. Each adf file usually starts with a configuration-frame, containing human readable information about the file structure, i.e. which frame types are present. If the structure of the file is known, that frame can be completely skipped. In the case of local level storage of all crystals, the configuration frame is followed exclusively by PSA frames until the end of the file. The PSA frame's contents are shown in table 4.4. There are other adf frames defined, but they are not relevant for the data analysis as presented in this work.

In general, it was tried to make the analysis in the simplest possible way, also with respect to the used software. One reason, why this was difficult is the fact that during the PreSPEC campaign, data was written in at least three different ways: separated AGATA and MBS data, merged AGATA and MBS data on the NARVAL side and merged AGATA and MBS data on the MBS side. That illustrates the severe lack of standards and documentation in the PreSPEC and AGATA collaboration. In view of the future use of AGATA at FAIR in the HISPEC experiments, the persons in charge of the project should take care to prevent these unnecessary complications.

offset (bytes)	size (bytes)	C type	meaning
0	2	uint16_t	AGATA crystal ID (number in the range 0 ... 179)
2	2	uint16_t	crystal status field
4	4	float	E0, energy from the low gain core preamplifier
8	4	float	E1, energy from the high gain core preamplifier
12	4	float	T0, time from the low gain core preamplifier
16	4	float	T1, time from the high gain core preamplifier
20	2	uint16_t	not used
22	2	uint16_t	N_{PSA} : number of PSA hits in current event
24	2	uint16_t	not used
26	2	uint16_t	not used
28	2	uint16_t	not used
N_{PSA} times the following data: $n = 0 \dots N_{\text{PSA}} - 1$			
$30 + n \cdot 42$	4	float	e : energy of the interaction point
$34 + n \cdot 42$	4	float	t : time of the interaction point (always 0)
$38 + n \cdot 42$	4	float	x : position of the interaction point
$42 + n \cdot 42$	4	float	y
$46 + n \cdot 42$	4	float	z
$50 + n \cdot 42$	4	float	Δe : corresponding uncertainties (always 0)
$54 + n \cdot 42$	4	float	Δt
$58 + n \cdot 42$	4	float	Δx
$62 + n \cdot 42$	4	float	Δy
$66 + n \cdot 42$	4	float	Δz
$70 + n \cdot 42$	2	uint16_t	not used

Table 4.4.: Content of the PSA frame header.

4.3.3 Coupling of AGATA to the VME

AGATA uses the GTS system for time stamp distribution and trigger validation. It is foreseen that AGATA is used together with VME based systems. In practice the VME system is connected to the GTS tree with a custom VME module: AGATA Ancillary VME Adapter (AGAVA) [22, section 7]. The AGAVA module can be a leaf in the GTS tree. At the same time it is controlled and read-out from the host VME system, MBS in this case. The front panel allows to sent trigger requests to the GTS root node. In the case of the PreSPEC setup, the validated MBS trigger was sent to the AGAVA module and the GTS root node was configured to accept all triggers from AGAVA.

Offline data merging

As data from each AGATA crystal and from the MBS system are recorded in separate files, the triggered event has to be build using the GTS timestamp that is available in all of the separate data files. The merging algorithm is described in the following.

From all available sources, the next event is read and the timestamps from all these events are compared. The smallest of these timestamps defines the beginning of a time window of a

given width of typically $3 \mu\text{s}$. All other read events that have a timestamp which falls inside this window, are used to build the merged event. All sources that contributed to the merged event are reading their next event. Now, the process repeats, and the smallest timestamp of all events is searched. This algorithm assumes, that the timestamps of events in each source are increasing, and that each source contributes only with multiplicity one or zero to any event.

Particle- γ time difference

It is essential to correlate the time of measured photons to a particle in the reaction chamber. This indicates, that γ radiation is actually coming from the projectile and not from a background source. The AGATA data acquisition (NARVAL) and MBS are only loosely coupled by distributed GTS timestamps with 10 ns resolution. For the particle- γ time correlation, a higher timing accuracy is desired. Therefore, the particle time T_p signal is measured with respect to the arrival of the GTS timestamp $T_{\text{GTS(MBS)}}$ in the MBS system: $\Delta t_{\text{GTS(MBS),p}} = T_p - T_{\text{GTS(MBS)}}$. In the AGATA system, the γ time $\Delta t_{\text{GTS(AGATA),}\gamma} = T_\gamma - T_{\text{GTS(AGATA)}}$ is already given with respect to the GTS timestamp $T_{\text{GTS(AGATA)}}$. The particle- γ time difference can then be calculated from the four quantities inside each event as

$$\Delta t_{p,\gamma} = T_{\text{GTS(AGATA)}} - T_{\text{GTS(MBS)}} + \Delta t_{\text{GTS(AGATA),}\gamma} - \Delta t_{\text{GTS(MBS),p}}. \quad (4.37)$$

In the data set, no time-walk correction was performed online during the experiment. The time resolution is thus not optimal yet. A typical fast in-beam energy-time correlation plot can be seen in Fig. 5.7. All important distinct structures are labeled. It is important to note that the major contribution of background radiation is induced by the projectile impact on the target, or detector material in proximity to the target. A narrow time gate can only distinguish between the prompt, the delayed beam induced, and the random γ -rays. The prompt background that is created at the exact same time as the γ -rays of interest from the projectile and target nucleus cannot be removed from the final spectrum. In this particular case the random background was increased by the presence of a ^{60}Co source, which can be subtracted from the data.

5 Analysis of the S424 experiment

The motivation of the S424 experiment was to test the performance of the complete PreSPEC setup by measuring known reactions in two types of experiments: Coulomb excitation and secondary beam fragmentation. In this section, the process of analyzing data from this experiment is described. Since this was no experiment to measure any unknown nuclear quantity, the emphasis is to describe a consistent picture of the physics processes involved during the experiment. A quantitative analysis of the Coulomb excitation cross-section ratio of the projectile and target excitation is presented. The beam was chosen to be equivalent to the primary beam ^{80}Kr in order to remove the FRS from the analysis and focus on the novelties of the setup. Furthermore, the lifetime of the first excited 2_1^+ state is 8.3 ps, which is not too short so that the decay happens primarily after leaving the target at the velocity as measured by LYCCA, and it is not too long so that the decay vertex is not too much behind the target (in average 1.2 mm for $v = 0.5 c$).

During the PreSPEC experimental campaign, the number of AGATA crystals was increasing. The AGATA crystal arrangement for the S424 experiment can be seen in figure 5.1.

5.1 Experiment parameters

The analysis in this chapter is for a subset of runs of the S424 experiment. A ^{80}Kr primary beam from the SIS-18 synchrotron with on-spill intensities between ≈ 5000 part./s up to ≈ 15000 part./s. The spill length was ≈ 8 s and the spill period was ≈ 13.3 s. The FRS was tuned to the primary beam, focused on the 380 mg/cm^2 thick gold target at the S4 focal point. The beam energy at the target center was 156.15 MeV/u . In this chapter, two different runs will be considered: AGATA run 14 with corresponding lmd files 326-421, and AGATA run 20 with corresponding lmd files 449-466. During run 14, $1.058 \cdot 10^8$ particles were delivered to the setup in 10.6 hours of beam, corresponding to an average particle rate of 2756 particles per second. During run 20, $2.450 \cdot 10^8$ particles were delivered to the setup in 8.6 hours of beam,

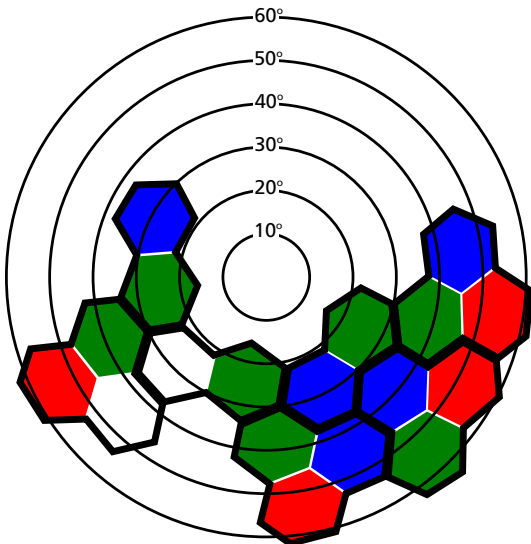


Figure 5.1.: The actual AGATA crystal arrangement in the S424 experiment as seen from the center of the target chamber, looking downstream. Polar angles are indicated up to 60° by the red circles. The center of the circles corresponds to the central beam axis. The crystals are arranged in double and triple cryostats, as indicated by the thick, black lines. Two cryostats, one double and one triple, had a missing A-type (red) crystal.

corresponding to an average particle rate of 7733 particles per second. The in-spill rate for both runs was about a factor of two higher than the average rate.

5.2 Qualitative description of processes in the target area

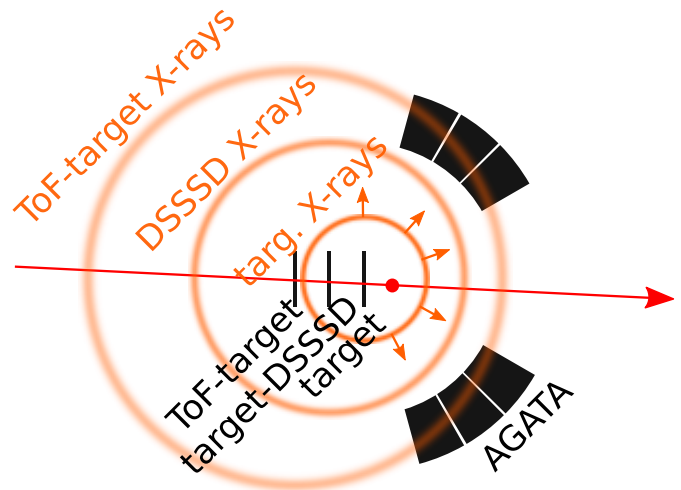
In this section, processes in the target area are qualitatively described, to motivate the event selection criteria that are later applied to the data in order measure the γ -ray yields.

One ^{80}Kr projectile enters the setup through the S41 scintillator and creates a particle trigger signal. That trigger signal indicates to the data acquisition, that a particle entered the setup. On its way to the gold target, the particle passes the LYCCA ToF-start detector, the entrance foil of the evacuated target chamber, the LYCCA DSSSD detector, and the LYCCA ToF-target detector. While passing all that matter, mostly atomic interactions between the projectile and the solid will happen. The term atomic interaction refers to the interaction between the beam nucleus and the electrons in solid matter. The probability for Coulomb excitation is low, because the detector material in front of the target is made out of low-Z material (Ti, Si, C and H). Sometimes, nuclear interactions between the projectile and detector or target nuclei happen and the projectile can be excited, fragmented or completely disintegrated, depending on the impact parameter of the collision. In Coulomb excitation experiments, a “good” event would be the excitation of the projectile inside the high-Z gold target due to electromagnetic interaction only. In secondary fragmentation experiments, a “good” event would be the fragmentation of the projectile inside a low-Z beryllium target.

Rutherford scattering and atomic interaction

Cross-sections for nuclear reactions are in general very small and most of the projectiles will pass the target material without any nuclear excitation. In this case, the projectile will have mainly interactions with the target electrons, creating atomic radiation, and with the nucleons via multiple Rutherford scattering, creating angular and energy straggling. The emitted radiation consists of low energy photons and scattered electrons. The photon spectrum is composed of: target X-rays, radiative electron capture (REC), primary bremsstrahlung (PB), secondary electron bremsstrahlung (SEB) and projectile X-rays, if the projectile is not fully stripped. The energy of emitted electrons is low enough to be stopped either inside the target or the vacuum

Figure 5.2.: Atomic background radiation created as the projectile passes through any kind of matter. In the case of Rutherford scattering, this is the only kind of radiation emitted. The time difference between the distinct flashes cannot be resolved by the AGATA detector. The average number N_{photon} of photons in each of the flashes depends on the projectile energy, the target material and the projectile isotope. In case of 150 MeV/u ^{80}Kr on the gold target $N_{\text{photon}} \approx 50$.



chamber, contributing to the SEB creation. How this kind of radiation is created in the PreSPEC setup is schematically shown in figure 5.2.

Coulomb excitation

In case of Coulomb excitation of even-even projectiles, the excitation process depends on the kinetic energy of the projectile: for low energies below $\approx 100 \text{ MeV/u}$, single step 2_1^+ excitation is the dominating process, while for energies above $\approx 300 \text{ MeV/u}$ excitation of the giant resonances dominates [82]. In case of excitation into the 2_1^+ state, only one single γ -ray will be emitted. If the transition energy is above the atomic background region, the γ -ray multiplicity above the atomic background region is expected to be one, the particle (neutrons or protons) multiplicity is expected to be zero. This process is schematically drawn in figure 5.3. In case of excitation into a giant resonance, the particle emission branching ratio is large. Therefore, these events can be identified by means of the particle identification in LYCCA. In the rare case of pure electromagnetic decay of a giant resonance, the γ -ray multiplicity with energies above the atomic background region will be larger than one.

Nuclear excitation

Nuclear excitation without particle knockout from the projectile can happen if the closest distance between projectile and target is below the range of nuclear interactions. This case looks similar to the genuine Coulomb excitation, only that the excitation mechanism is not purely electromagnetic. Instead of a virtual photon, a virtual pion is exchanged.

Fragmentation reactions

If the projectile passes the target such that the matter distributions are very close or even overlap, nuclear fragmentation occurs. At typical PreSPEC energies, this process can be described by quantum-mechanical molecular dynamic models [83]. In the collision, projectile and target get into a highly excited state, and the decay can be qualitatively separated in two processes. First, a dynamically determined, fast process, which happens before the system can equilibrate. In this process, many nucleons and some light clusters escape the system. Second, a statistical process, where the equilibrated residue evaporates nucleons to cool down. This process is schematically drawn in figure 5.4. Evaporated neutrons will be visible inside the HPGe detectors as $\text{Ge}(n, n'\gamma)\text{Ge}$ reactions. Charged particles will deposit a lot of energy inside the HPGe detec-

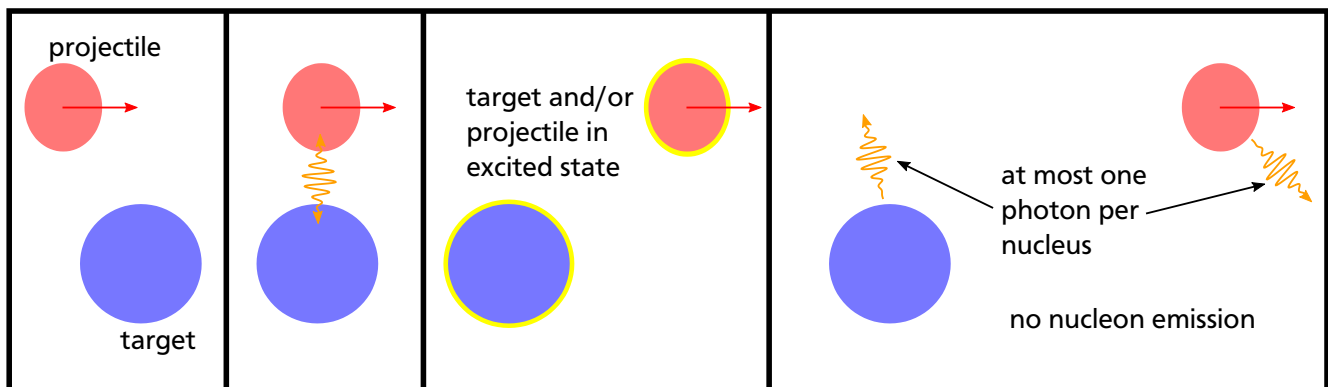


Figure 5.3.: Stages of the nuclear Coulomb excitation reactions in the case of excitation of the first excited state.

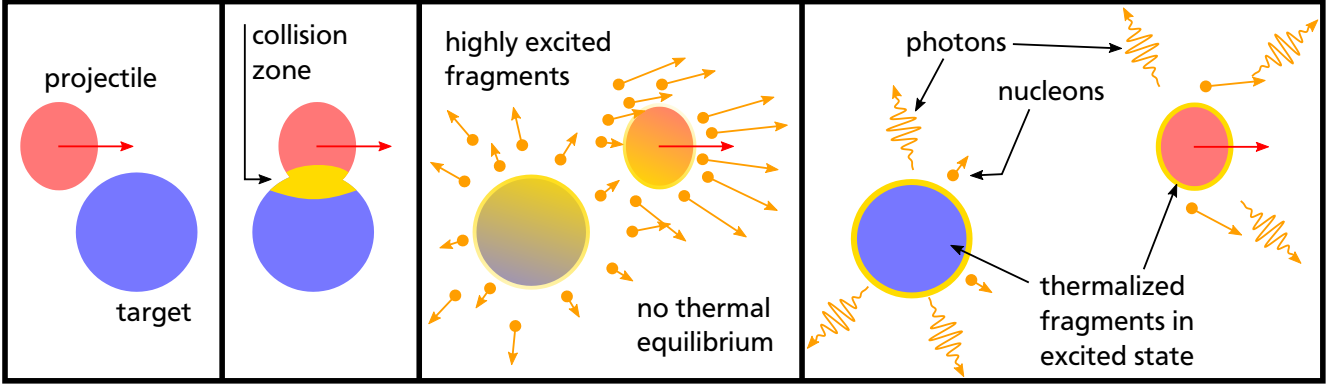


Figure 5.4.: Stages of the nuclear fragmentation reactions. The pictures are a qualitative and schematic reproduction of figure 36 in [83], where a gold-gold collision at 200 MeV/u is shown for different impact parameters.

tors and will always saturate the ADC. Particles emitted in small forward angles will hit LYCCA, but do not deposit enough energy to be detected if their charge number Z is below ≈ 10 . In case of very small collision parameters, or even central collisions, the resulting piece of nuclear matter is highly excited and undergoes nuclear multifragmentation [84]. The cross section for this process is very small, but in this case multiple intermediate mass fragments (IMF) can be seen in the LYCCA Wall. An indication that the charged particles seen by AGATA are actually coming from the nuclear fragmentation reaction, is the correlation between the lost mass of the projectile in the reaction and the multiplicity of ADC-saturating events in AGATA. Assuming that the main reason for AGATA ADC saturation is a charged particle, the multiplicity of that is approximately equal to the detected charged particle multiplicity. This correlation is shown in figure 5.5

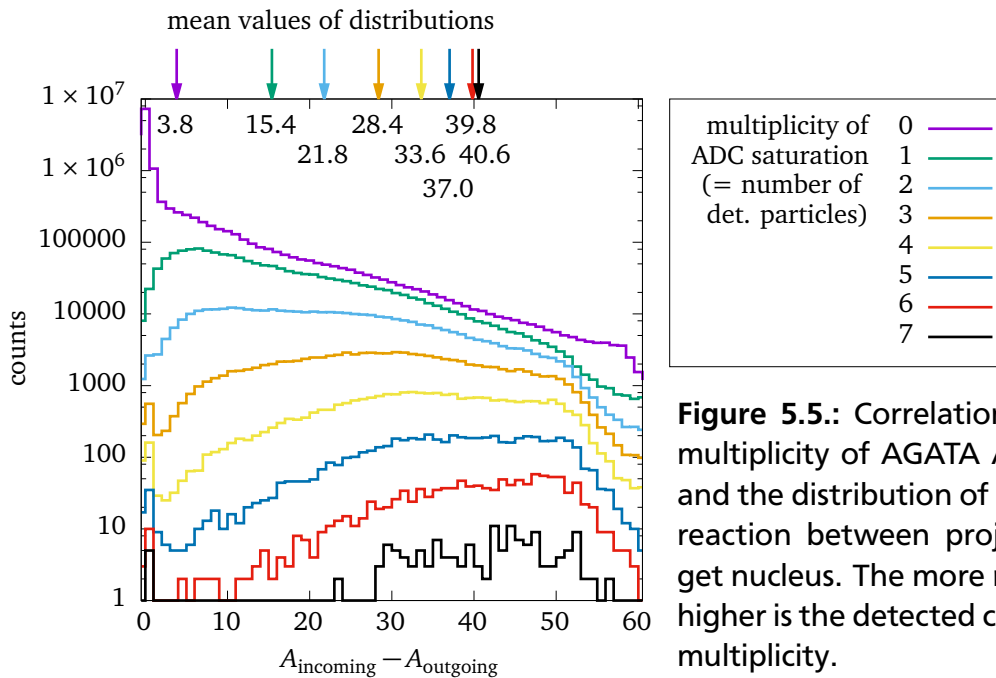


Figure 5.5.: Correlation between the multiplicity of AGATA ADC saturation and the distribution of lost mass in the reaction between projectile and target nucleus. The more mass is lost, the higher is the detected charged particle multiplicity.

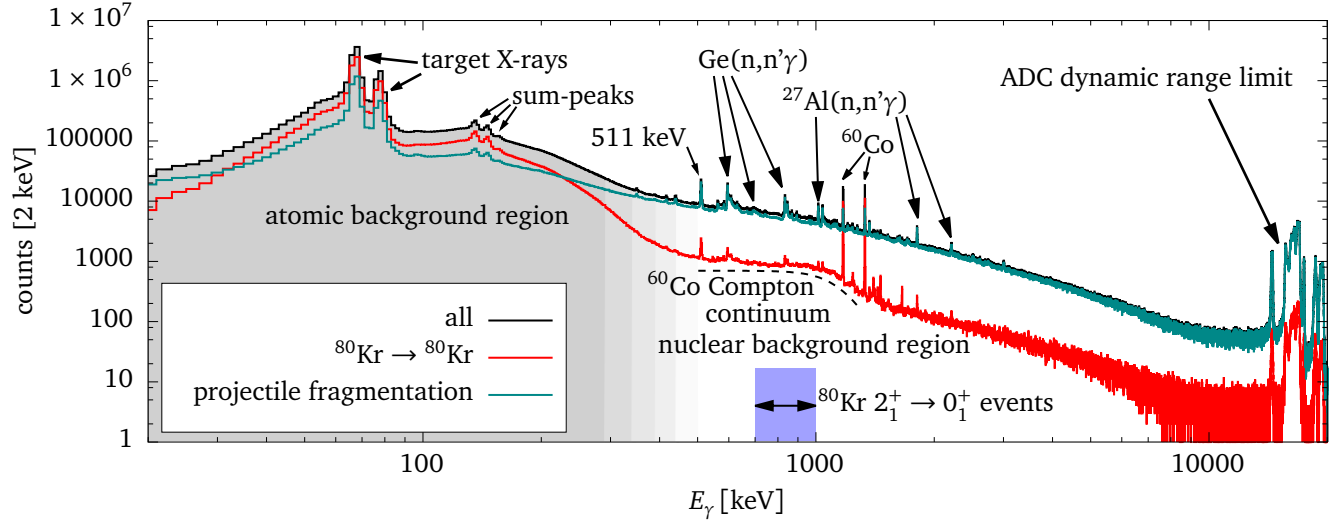


Figure 5.6.: Overall picture of the prompt γ -ray spectrum from AGATA for a 150 MeV/u ^{80}Kr beam on a 380 mg/cm² thick gold target, using the core signals, i.e. no γ -ray tracking. Both axis have logarithmic scaling to show all features.

Multi-hit events

It was observed, that more than one ^{80}Kr -like particle is passing through the target area within one triggered event. In such a case, there is no simple way of establishing a correlation between the detected γ -rays and the particles. Due to the high segmentation of the LYCCA wall detector, and the multi-hit capability of the LYCCA-ToF detectors, it may in principle be possible to disentangle these events and extract multiple good events out of a multi-hit event. The success of experiment S426 depends on the successful implementation of such a technique. In general, it was observed that the multi-hit event rate increased with the primary particle rate. In *all* other RISING or PreSPEC experiment analysis so far, all multi-hit events were rejected.

5.3 Properties of γ -ray spectra

A typical γ -ray spectrum from an in-beam experiment, recorded with HPGe detectors, can be seen in figure 5.6. The black histogram has no further condition applied, apart from the particle- γ trigger condition. The region below 500 keV, indicated by the gray background, is dominated by atomic background radiation, that is induced by the interaction of the projectile with the target electrons and from the X-ray radiation from the target atoms, as described in the previous subsection. Because of the high multiplicity of X-ray photons, sum energy peaks of two X-ray photons appear. The region from 700 keV up to 1000 keV, as indicated by the blue box, is the region of interest for the Coulomb excitation of ^{80}Kr . Because of the Doppler effect, radiation from the de-excitation of the 2_1^+ state at 616.6 keV shifted into that energy region. This region is dominated by peaks that are induced by $\text{Ge}(n, n'\gamma)$ and $\text{Al}(n, n'\gamma)$ (contained in the germanium encapsulation) reactions, indicating a high neutron flux in the proximity of the AGATA detectors. These neutrons are created in nuclear reactions of projectile and target nuclei, which is why these peaks are largely suppressed if the projectile stays intact, as shown by the red spectrum. Also protons are created in these nuclear reactions, but they deposit much more energy inside the germanium crystal. These signals are in general outside the dynamic

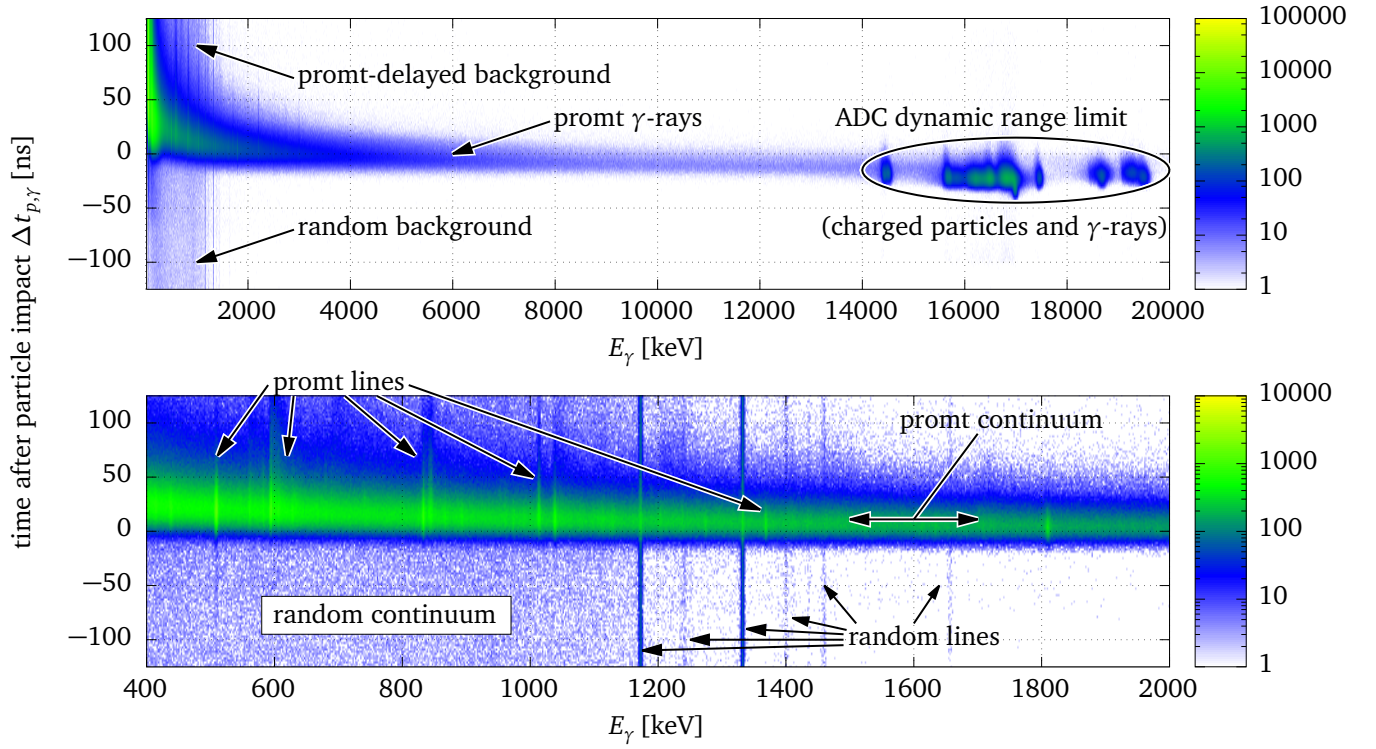


Figure 5.7.: Correlation plot of AGATA particle- γ time difference and AGATGA γ energy. The upper panel shows the full dynamic energy range of the AGATA FEE. The lower panel is a zoom to the most relevant energy region for in-beam γ spectroscopy. Signatures of different features in the spectrum are labeled. Note that the random continuum is rather small compared to the prompt background and is, in this case, dominated by the Compton continuum of the ^{60}Co source that was present during this experiment.

range of the ADC in the digitizer, and the corresponding energy events accumulate at the end of the spectrum. The end of the dynamic range was different for each detector, thus each peak corresponds to one detector. Also for these peaks, a drastic reduction is observed when applying the condition of having no fragmentation of the projectile, because the main source of high energy deposition in the AGATA crystals are charged particles from fragmentation reactions. Figure 5.7 shows a correlation plot of the γ energy and the particle- γ time difference.

5.4 Doppler Effect

The projectile velocity in PreSPEC-like experiments is large enough to cause a significant Doppler effect on the emitted γ -rays. For the emission of γ -radiation, the Doppler effect causes a change in energy and emission angle with respect to the velocity vector. It can be calculated by applying a Lorentz boost to the momentum 4-vector of a photon (p^μ) = ($E/c, \vec{p}$). For a photon with rest mass $m = 0$, the energy momentum relation is $E = c |\vec{p}|$, and the momentum 4-vector can be written as (p^μ) = $|\vec{p}| (1, \vec{p}/|\vec{p}|)$. In a cylindrical coordinate system ($\vec{e}_z, \vec{e}_\rho, \vec{e}_\phi$), in which the nucleus moves along the z -direction, the photon momentum can be written as

$\vec{p} = |p| (\vec{e}_z \cdot \cos \vartheta + \vec{e}_\rho \cdot \sin \vartheta)$. Applying the Lorentz boost matrix to the 4-vector in the laboratory frame $\Lambda^\mu{}_\nu p_{\text{lab}}^\nu = p_{\text{rest}}^\mu$, yields the 4-vector in the rest frame of the nucleus

$$\frac{E_{\text{lab}}}{c} \begin{pmatrix} \gamma & 0 & -\beta \gamma \\ 0 & 1 & 0 \\ -\beta \gamma & 0 & \gamma \end{pmatrix} \begin{pmatrix} 1 \\ \sin \vartheta_{\text{lab}} \\ \cos \vartheta_{\text{lab}} \end{pmatrix} = \frac{E_{\text{lab}}}{c} \begin{pmatrix} \gamma (1 - \beta \cos \vartheta_{\text{lab}}) \\ \sin \vartheta_{\text{lab}} \\ \gamma (\cos \vartheta_{\text{lab}} - \beta) \end{pmatrix} = \frac{E_{\text{rest}}}{c} \begin{pmatrix} 1 \\ \sin \vartheta_{\text{rest}} \\ \cos \vartheta_{\text{rest}} \end{pmatrix}, \quad (5.1)$$

from which the two known equations for the change in energy and angle can be obtained:

$$E_{\text{rest}} = E_{\text{lab}} \frac{1 - \beta \cos \vartheta_{\text{lab}}}{\sqrt{1 - \beta^2}} \quad (5.2)$$

$$\cos \vartheta_{\text{rest}} = \frac{E_{\text{lab}}}{E_{\text{rest}}} \frac{\cos \vartheta_{\text{lab}} - \beta}{\sqrt{1 - \beta^2}} = \frac{\cos \vartheta_{\text{lab}} - \beta}{1 - \beta \cos \vartheta_{\text{lab}}}. \quad (5.3)$$

If radiation is emitted in the rest frame of a moving nucleus with a distribution $W_{\text{rest}}(\vartheta_{\text{rest}})$, this distribution is transformed into another distribution seen in the laboratory frame $W_{\text{lab}}(\vartheta_{\text{lab}})$:

$$W_{\text{rest}}(\vartheta_{\text{rest}}) \sin \vartheta_{\text{rest}} d\vartheta_{\text{rest}} = W_{\text{rest}}(\vartheta_{\text{rest}}(\vartheta_{\text{lab}})) \sin(\vartheta_{\text{rest}}(\vartheta_{\text{lab}})) \frac{d\vartheta_{\text{rest}}}{d\vartheta_{\text{lab}}} d\vartheta_{\text{lab}} \quad (5.4)$$

$$= \underbrace{W_{\text{rest}}(\vartheta_{\text{rest}}(\vartheta_{\text{lab}})) \frac{1 - \beta^2}{(1 - \beta \cos \vartheta_{\text{lab}})^2}}_{:= W_{\text{lab}}(\vartheta_{\text{lab}})} \sin \vartheta_{\text{lab}} d\vartheta_{\text{lab}}, \quad (5.5)$$

and after inserting equation (5.3), the final equation is obtained

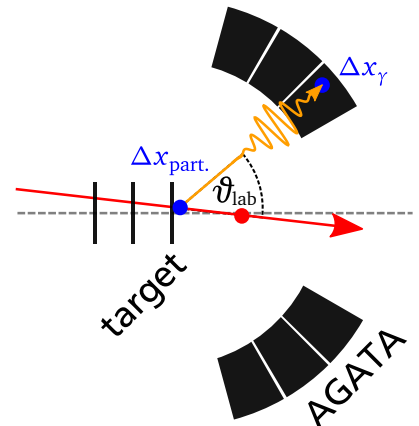
$$W_{\text{lab}}(\vartheta_{\text{lab}}) = W_{\text{rest}} \left(\arccos \left[\frac{\cos \vartheta_{\text{lab}} - \beta}{1 - \beta \cos \vartheta_{\text{lab}}} \right] \right) \frac{1 - \beta^2}{(1 - \beta \cos \vartheta_{\text{lab}})^2}. \quad (5.6)$$

Doppler Correction

The continuous position sensitivity of AGATA allows to see the Doppler Effect at different polar angles θ of γ -ray emission. The polar angle is measured with respect to the outgoing particle velocity vector, as shown in figure 5.8.

The Doppler effect has an impact on the achievable energy resolution of mono-energetic γ -rays. After correction of the Doppler shift, the resulting γ -ray energy in the nuclear rest frame

Figure 5.8.: Polar angle of γ -ray emission with respect to the projectile velocity vector. Its uncertainty is determined by the position uncertainties Δx_γ and Δx_{part} .



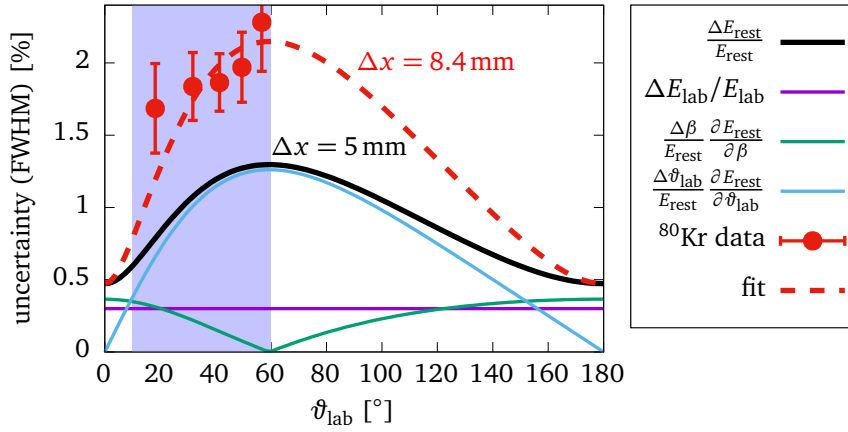


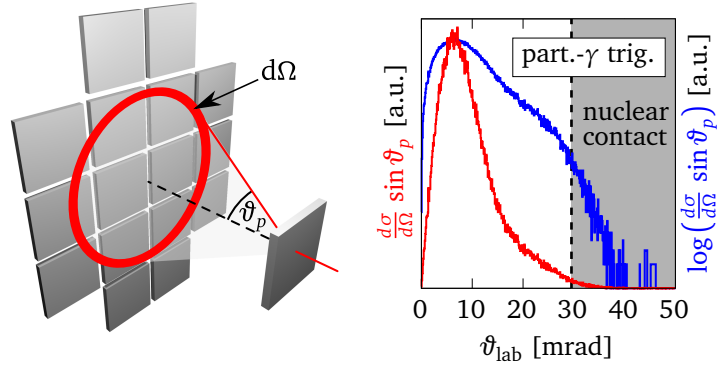
Figure 5.9.: Resolution of Doppler-corrected γ -ray detection for $\Delta\beta/\beta = 0.53\%$, $\beta = 0.51$, $\Delta\vartheta_{lab} = \Delta x/235$ mm. The light blue angular range is covered by AGATA. The fit was done varying $\Delta\vartheta_{lab}$, which was used to deduce a combined uncertainty for projectile position and γ location of 8.4 mm.

depends on the three quantities E_{lab} , ϑ_{lab} and β . They are measured with uncertainties ΔE_{lab} , $\Delta\vartheta_{lab}$ and $\Delta\beta$. By Gaussian propagation of errors one gets for the width of the Doppler corrected γ -ray energy

$$\begin{aligned} \Delta E_{rest} &= \sqrt{\left(\Delta E_{lab} \frac{\partial E_{rest}}{\partial E_{lab}}\right)^2 + \left(\Delta\beta \frac{\partial E_{rest}}{\partial \beta}\right)^2 + \left(\Delta\vartheta_{lab} \frac{\partial E_{rest}}{\partial \vartheta_{lab}}\right)^2} \\ \Rightarrow \frac{\Delta E_{rest}}{E_{rest}} &= \sqrt{\left(\frac{\Delta E_{lab}}{E_{lab}}\right)^2 + \left(\frac{\Delta\beta}{E_{rest}} \frac{\partial E_{rest}}{\partial \beta}\right)^2 + \left(\frac{\Delta\vartheta_{lab}}{E_{rest}} \frac{\partial E_{rest}}{\partial \vartheta_{lab}}\right)^2} \end{aligned} \quad (5.7)$$

Figure 5.9 shows the impact of the three contributing terms to the total uncertainty of the Doppler corrected energy. For the velocity uncertainty, the upper limit from section 4.1.3 was taken. The LYCCA ToF measurement was well tuned and cannot be improved. The peak width, as shown by the black line in figure 5.9, is dominated by the angular uncertainty, which was assumed to be due to a 5 mm (FWHM) uncertainty in the location of the first interaction point in AGATA at the nominal AGATA distance of 235 mm, which is the inner radius of the AGATA detector sphere. The actual resolution is much worse, as can be seen by the data points that were determined at 5 different observation angles. A fit to the data with $\Delta\vartheta_{lab}$ as free parameter indicates, that the position uncertainty was $\Delta x = 8.4$ mm, again assuming the nominal inner AGATA radius. This value is the combined uncertainty from the interaction location of the γ -ray in AGATA Δx_γ , and the uncertainty of the projectile decay position $\Delta x_{part.}$. A PSA replay of AGATA data would probably increase the position resolution of the interaction points, which is one of the contributions to the uncertainty $\Delta\vartheta_{lab}$. The main contributions to $\Delta x_{part.}$ are systematical errors in the target-detector geometry. It was shown by Stahl [66], that an optimization of the geometry with respect to the γ peak resolution can be done. In such an optimization, the Doppler correction is repeated for different offsets (x, y, z) of the target location is varied such that the peak resolution is minimal. Both optimizations have to be done in the future, in order to determine the achievable resolution of AGATA in the PreSPEC setup, but are outside the scope of the present work.

Figure 5.10.: Schematic drawing of the LYCCA Wall detector modules and the target (not to scale). The differential solid angle $d\Omega$ is increasing with the particle scattering angle as $\sin\vartheta_p$. This has to be taken into account when comparing measured differential cross-sections to calculated curves of $\frac{d\sigma}{d\Omega}$.



5.5 Scattering angle and angular straggling

The particle scattering angle can be used to determine the impact parameter of the collision. For experiments with a collision energy above the Coulomb barrier, this information can be used to exclude collisions with nuclear contact. In addition to the nuclear scattering process, the observed scattering angle is influenced by angular straggling that always happens when an ion passes matter as a result of multiple Rutherford scattering of the ion on the target atoms (see [85]). The observed scattering angle distribution is the convolution of the nuclear scattering distribution and the angular straggling distribution. For any process, the angular differential cross-section is typically given as $\frac{d\sigma}{d\Omega}(\vartheta_p)$. If this quantity is measured with the PreSPEC (or any similar) setup, one has to take into account that the particle detection efficiency is increasing with increasing particle angle ϑ_p , because of the full coverage of all relevant solid angles by the particle detectors. Consequently, the particle detection system measures the quantity $\frac{d\sigma}{d\Omega}(\vartheta_p) \sin\vartheta_p$. This is visualized in figure 5.10, together with a histogram of particle scattering angles with a part.- γ trigger condition. For comparison with theoretical predictions, the calculated differential cross-section has to be multiplied with $\sin\vartheta_p$. For the two mentioned reasons, angular straggling and the $\sin\vartheta_p$ term, the scattering angle histogram in figure 5.10 is always zero at $\vartheta_p = 0$. It also can be seen, that the particle scattering angle drops off at $\vartheta_p = 29.5$ mrad which is the grazing angle of the reaction, i.e. the distance of closest approach at this angle is identical to the sum of the two nuclear radii. No particles are detected at angles beyond 40 mrad.

5.6 Coulomb excitation of ^{80}Kr

All data shown in this section is from the AGATA runs 14 and 20 and corresponding lmd files of the S424 experiment.

5.6.1 Background reduction

From the large amount of recorded data, only a tiny fraction was caused by the process of interest, which is in this case Coulomb excitation of the ^{80}Kr projectile and the ^{197}Au target. To get an idea of the strength of this particular signal, the non-Doppler corrected signal is drawn as a blue square in figure 5.6. The first step in analyzing such data is the definition of criteria to classify and select the subset of events that contribute to the signal of interest. Based on

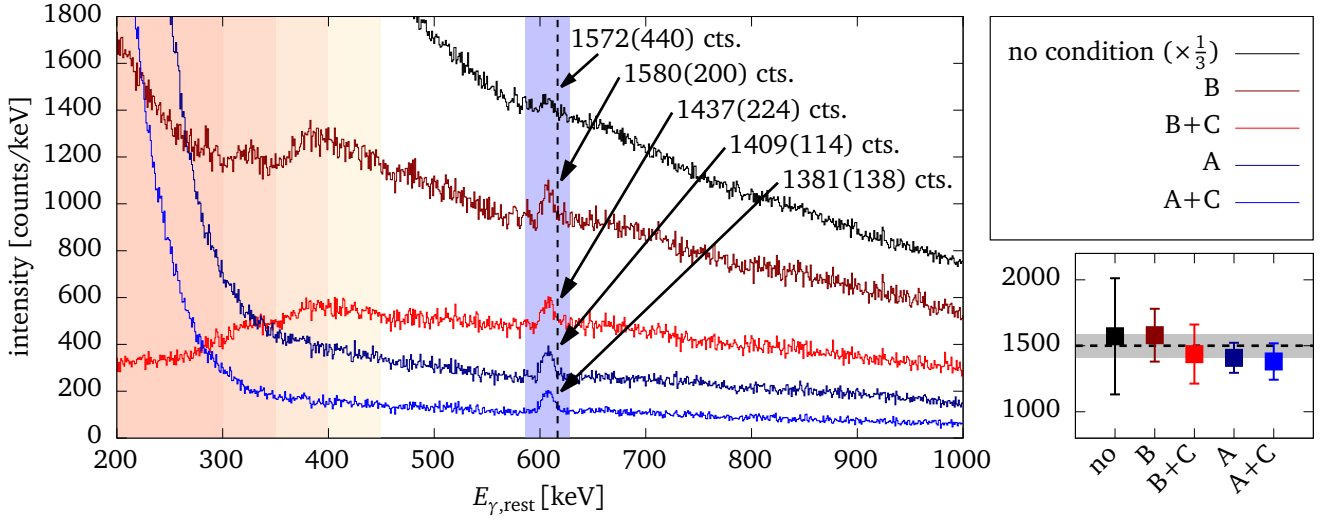


Figure 5.11.: Doppler corrected γ -ray spectra under different background subtracting conditions: A \rightarrow LYCCA identification of ^{80}Kr , B \rightarrow Multiplicity of γ -rays above 450 keV (in the laboratory frame) equals one, C \rightarrow time gate of 22 ns around the prompt time. No significant impact on the peak area is observed. The unconditional spectrum is scaled by a factor of $1/3$. The peak area was measured with a Gaussian least-squares fit and linear background in the light blue region around the peak. A more rigorous determination of the final peak area is shown in figure 5.12, the value is indicated by the gray band and dashed line in the bottom left. The red area indicates the region of energies that does not contribute to the multiplicity count. In that region, the γ -multiplicity condition distorts the shape of the spectrum. The dashed line shows the excitation energy of the first 2_1^+ state in ^{80}Kr at 616.6 keV.

the previous discussion, there are three main criteria for the presence of a Coulomb excitation process in the target chamber: only one single ion was present in the target area, only one single γ -ray (above atomic background energies) and no particle was emitted in the projectile target interaction, and the projectile was identified by LYCCA as ^{80}Kr , i.e. no fragmentation took place. The primary particle multiplicity can be seen in any segmented detector, such as the LYCCA CsI-wall, or in any detector with multi-hit capability, such as the LYCCA ToF-detectors. Multiplicity one is required in both detector systems. The γ -ray multiplicity criterion can be realized by counting the number of γ -rays in given window inside a window similar to figure 5.7, but going up to the high end of the γ -ray energy. The LYCCA identification criterion is the identification of $A = 80$ and $Z = 36$.

In addition, a prompt time gate is used and a random background subtraction is done. In this particular case, the main source of random background in the relevant energy region was the ^{60}Co source during the runs. This can be seen by the difference of the region above and below the two relatively strong ^{60}Co lines (1173 keV and 1332 keV) in figure 5.7. The effect of LYCCA identification and γ -multiplicity condition on the background reduction are shown in figure 5.11 for the Doppler corrected spectrum.

5.6.2 Doppler corrected γ spectra

Applying all mentioned conditions and doing a random particle- γ time background subtraction, spectra as shown in figure 5.12 can be obtained. Being able to observe both, target and projectile excitation due to the Coulomb interaction allows to calculate the Coulomb excitation cross-section ratio of the two nuclei. Their level-schemes are shown in figure 5.12 and the excited state in Coulomb excitation reactions are indicated. Before comparing both γ -ray yields, the different efficiency due to the angular distribution and energy of the γ -rays has to be taken into account.

5.6.3 Angular distribution of γ -rays

In order to get an estimate of the influence of the γ -ray angular distribution on the detection efficiency, the angular distribution was calculated according to the formulas found in [88, B.3]. The angular distribution is based on the analytic formulas for single step Coulomb excitation in the straight-line approximation according to [31] from an initial state I_i to a final state I_f and the subsequent γ transition into a state I_{ff} :

$$W(\vartheta_\gamma) = \sum_{k \text{ even}, \mu, L, L'} \left| G_{E\lambda\mu} \left(\frac{c}{v} \right) \right|^2 g_\mu(\xi(b_{\min})) (-)^\mu \begin{pmatrix} \lambda & \lambda & k \\ \mu & -\mu & 0 \end{pmatrix} \times \begin{Bmatrix} I_f & I_f & k \\ \lambda & \lambda & I_i \end{Bmatrix} F_k(L, L', I_{ff}, I_f) \sqrt{2k+1} P_k(\cos(\vartheta)) \quad (5.8)$$

where the functions $G_{E\lambda\mu}$ and K_μ are the same as in section 3.3.3. The function $g_\mu(\xi(b_{\min}))$ is

$$g_\mu(\xi(b_{\min})) = \pi \xi^2 \left(\left| K_{\mu+1}(\xi(b_{\min})) \right|^2 - \left| K_\mu(\xi(b_{\min})) \right|^2 - \frac{2\mu}{\xi} K_{\mu+1}(\xi(b_{\min})) K_\mu(\xi(b_{\min})) \right). \quad (5.9)$$

The definition of $F_k(L, L', I_{ff}, I_f)$, as well as for the Wigner-3j and Wigner-6j symbols, can be found for example in [89]. The formula is valid for pure $E2$ excitation and de-excitation. From the solid angle coverage of the AGATA crystals, one can obtain a distribution of geometric efficiency of AGATA. The geometric efficiency $\varepsilon_{\text{geo}}(\vartheta_{\text{lab}})$, the angular distribution $W(\vartheta_{\text{lab}})$, and known values for the energy dependence of the AGATA efficiency $\varepsilon(E_\gamma)$ for γ -radiation (e.g. from [75]), an efficiency ratio of target and projectile detection can be calculated:

$$\begin{aligned} \varepsilon_{\text{tot.}} &\propto \int_0^\pi \varepsilon_{\text{geo}}(\vartheta_{\text{lab}}) W(\vartheta_{\text{lab}}) \varepsilon_{E_\gamma}(E_\gamma(\vartheta_{\text{lab}})) \sin \vartheta_{\text{lab}} d\vartheta_{\text{lab}} \\ \Rightarrow \frac{\varepsilon_{\text{proj.}}}{\varepsilon_{\text{targ.}}} &= \frac{0.0247}{0.0124} = 1.99. \end{aligned} \quad (5.10)$$

The total efficiency $\varepsilon_{\text{proj.}}$ for detection of projectile γ -rays will later be used to estimate the total efficiency of the setup. The total efficiencies were numerically calculated based on the data

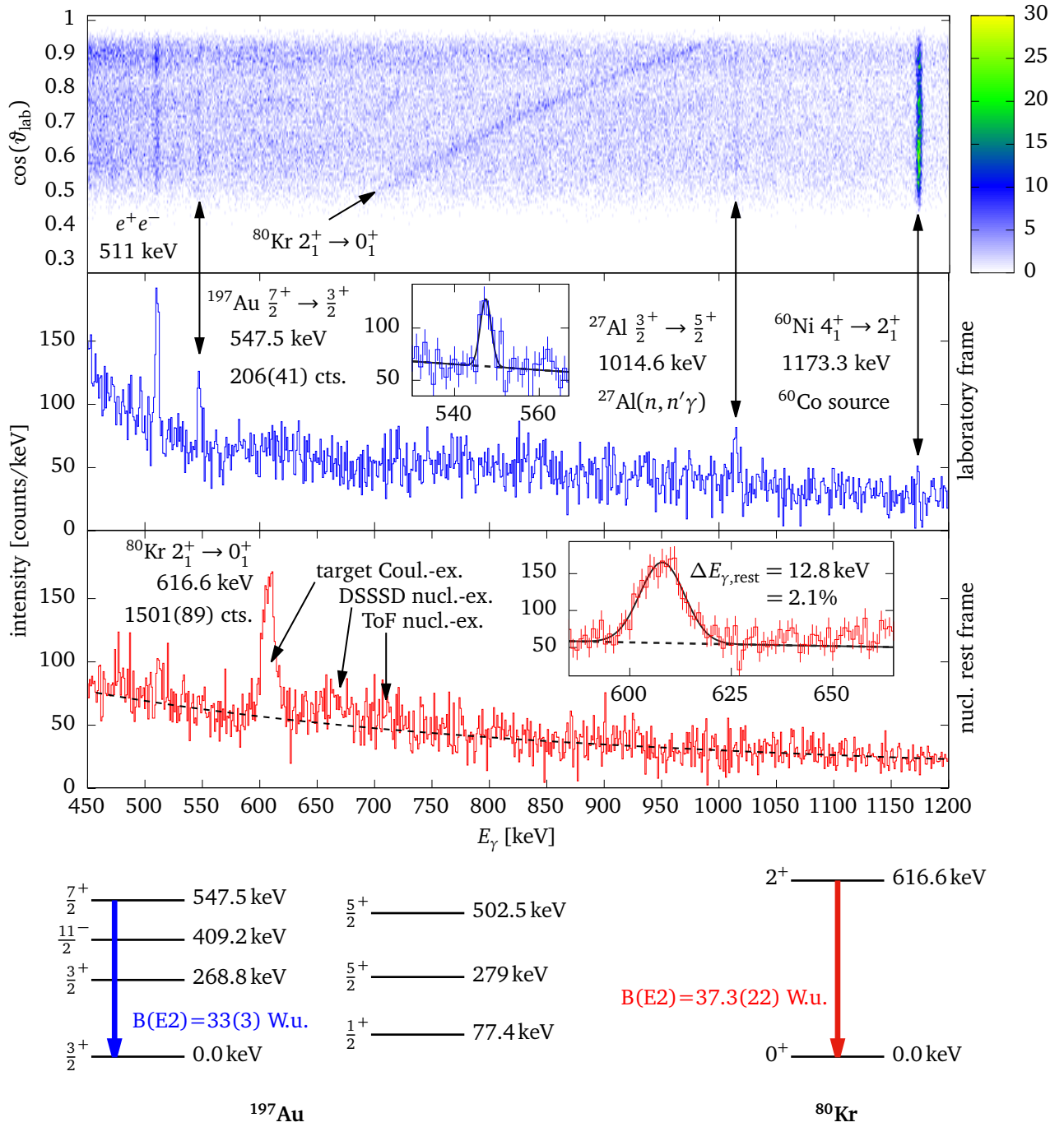


Figure 5.12.: Gated AGATA γ -ray spectra. The top panel (no time background subtraction) shows the correlation of γ -ray emission angle ϑ_{lab} and the energy. Doppler shifted lines, emitted by flying particles, are related to the emission angle as in equation (5.2). The 1173.3 keV line from the ^{60}Co source is also visible. The lower two panels show energy spectra in the laboratory (middle) and projectile rest frame (bottom) with a random time background subtraction applied. In the middle panel, the ^{60}Co line is not visible anymore. Only 511 keV electron-positron annihilation line and the target excitation at 547.5 keV, and only one background line from an $(n, n'\gamma)$ reaction is visible. In the Doppler corrected spectrum, only the projectile excitation is visible. The excited state in a Coulomb excitation experiment will be the lowest lying one with a large $E2$ matrix element to the ground state. Excitation on the ToF-target and DSSSD detectors are discussed in section 5.6.8. Level scheme data from [86] and [87].

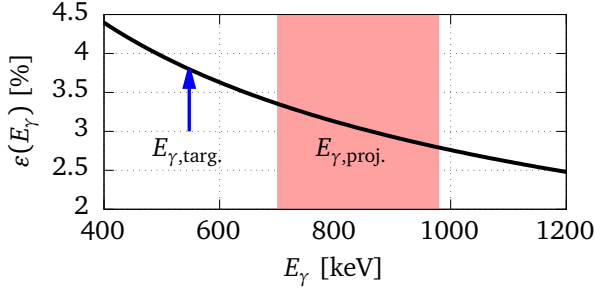


Figure 5.13.: The AGATA efficiency curve, as used in the calculation of the absolute and relative γ -ray efficiencies. The parametrization is: $\varepsilon = \exp(D + E x + F x^2)/N$, with $x = \log(E_\gamma/1000)$, and $D = 6.410$, $E = -0.573$, $F = -0.071$, and $N = 0.00454$.

shown in figure 5.14. Energy dependent AGATA efficiency curves were taken from the core common parameterization of [75], reproduced in figure 5.13. This efficiency curve was normalized by the integral of the geometric efficiency and scaled by the factor $16/21$, because the measurements in [75] were done with 21 AGATA crystals in the setup, while in S424 experiment only 16 crystals were mounted. An error estimate was not done, because uncertainties are entirely dominated by the uncertainty of experimental peak areas. The resulting correction factor for

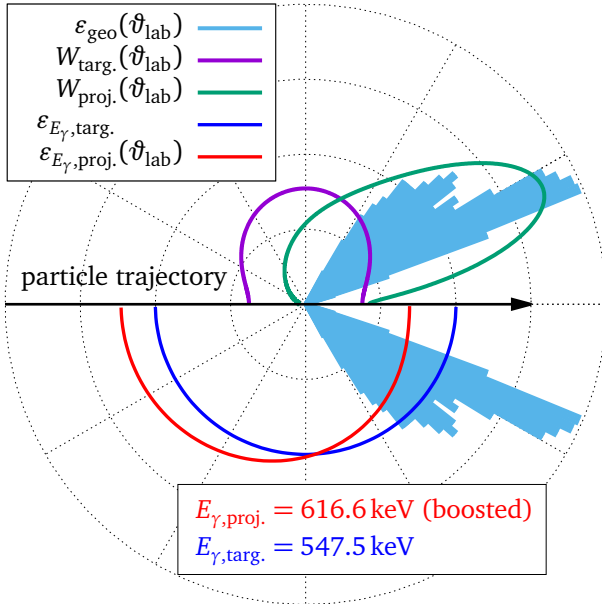


Figure 5.14.: Upper half: angular distribution of γ -radiation from target (magenta) and projectile (green), measured with respect to the particle trajectory. The calculation was done using equations (5.8) and (5.6). Lower half: AGATA single crystal photo peak efficiency for γ -ray detection for target (blue) and projectile (red) radiation. AGATA azimuthal angular coverage (light blue) was extracted from the AGATA crystal arrangement pattern (see figure 5.1), and is shown on the upper and lower halves to see the relevant parts of the other distributions. All magnitudes are in arbitrary units.

the efficiency can be used to calculate the cross-section ratio of target and projectile excitation, with peak areas $A_{\text{proj.}} = 1501(89)$ and $A_{\text{targ.}} = 206(41)$, respectively:

$$\frac{\sigma_{\text{proj.}}}{\sigma_{\text{targ.}}} = \frac{A_{\text{proj.}}}{A_{\text{targ.}}} \cdot \frac{\varepsilon_{\text{targ.}}}{\varepsilon_{\text{proj.}}} = 3.66(73). \quad (5.11)$$

The efficiency calculation contains the following simplifications, which can only be avoided by using a detailed (Monte-Carlo) simulation of the response of the setup with Geant4:

- The dependence of the AGATA efficiency on the decay position of the projectile is not taken into account.
- The γ -ray angular distribution was calculated with respect to the z-axis, not taking into account the directional distribution of the outgoing particles.
- No efficiency calibration data was used, but only a combination of crystal efficiency and solid angle coverage.

	parameter	target	projectile
kinematic	Z	79	36
	A	197	80
	$E_{\text{inc.}}$	-	156.15 MeV/u
excitation	$I_{\text{g.s.}}$	$\frac{3}{2}$	0
	$I_{\text{ex.}}$	$\frac{7}{2}$	2
	$E_{\text{ex.}}$	0.5475 MeV	0.6166 MeV
	$\langle f \hat{M}_{E2} i \rangle$	134.01 efm ²	61.8 efm ²
opt. potential	r_0	1.2 fm	1.2 fm
	$r_{0,I}$	1.2 fm	1.2 fm
result	$\sigma_{\uparrow, \text{clx.}}$	108 mb	442 mb

Table 5.1.: Parameters for the differential cross-section calculation with DWEIKO. All mentioned parameters have been changed with respect to the default parameters in the example input file. Only one excitation was included in the calculation because multi-step processes are practically not happening at these energies.

5.6.4 Comparison with theory

For the projectile energy in the DWEIKO calculation, 156.15 MeV/u was used as an estimate of the energy in the middle of the target, based on the outgoing velocity and an ATIMA calculation. The calculation of the nuclear $E2$ matrix element was based on the adopted $B(E2; \downarrow)$ values (see figure 5.12). One Weisskopf unit (W.u.) is defined as

$$1 \text{ W.u.} = \frac{1}{4\pi} \left(\frac{3}{\lambda + 3} \right)^2 e^2 R_0^{2\lambda} = \begin{cases} 20.5 e^2 \text{fm}^4 & \text{for } {}^{80}\text{Kr} \\ 68.1 e^2 \text{fm}^4 & \text{for } {}^{197}\text{Au} \end{cases} \quad (5.12)$$

with $R_0 = 1.2 \text{ fm} \cdot A^{1/3}$ and $\lambda = 2$ for $E2$ transitions. The transition matrix element is calculated by the relation to the $B(E2)$ value

$$B(E2; I_i \rightarrow I_f) = \frac{1}{2I_i + 1} |\langle f || \hat{M}_{E2} || i \rangle|^2 \quad (5.13)$$

With the initial I_i and final I_f spins of the connected states. A summary of all relevant DWEIKO parameters is given in table 5.1. The resulting theoretical cross section-ratio between projectile and target is

$$\frac{\sigma_{\text{proj.}}}{\sigma_{\text{targ.}}} = \frac{442 \text{ mb}}{108 \text{ mb}} = 4.09 \quad (\text{DWEIKO}) \quad (5.14)$$

$$3.66(73) \quad (\text{experiment}), \quad (5.15)$$

which is in agreement with the experiment. Nuclear excitation is not taken into account. The main source of uncertainty is the small number of counts in the γ peak of the target excitation, which is mainly due to the lack of detectors at 90° , where the angular distribution of the target de-excitation is large.

5.6.5 Differential cross-section

The differential cross-section $\frac{d\sigma}{d\Omega}$ can be extracted from the measured data to compare it to the DWEIKO calculation. The theoretical differential cross-section does not include any angular

straggling of the projectile in the target material. In order to compare it with experimental data, the angular straggling has to be taken into account. The magnitude of the angular straggling $\sigma_{\text{target}} = 5.343 \text{ mrad}$ was calculated with ATIMA [58] for the 380 mg/cm^2 gold target. The contributions from the $300 \mu\text{m}$ thick Si-detector and the $500 \mu\text{m}$ thick LYCCA ToF-target detector are small. Also, the additional matter further upstream contributes only little, because of the long lever to the scattering point. In the end, a total value of $\sigma_{\text{strg.}} = 5.5 \text{ mrad}$ was used and folded into the differential cross section curve from DWEIKO with a Monte-Carlo method that is described in the following.

Values of polar angles ϑ were drawn randomly from the distribution $\frac{d\sigma}{d\Omega} \sin \vartheta$ from DWEIKO. Together with a uniformly distributed azimuthal angle φ , a direction vector was calculated. The direction was then modified by drawing another pair of (ϑ, φ) angles, where the polar angle was following a Gaussian distribution with width $\sigma_{\text{strg.}}$. The direction vectors were combined and the polar angle of the resulting vector was calculated. The resulting distribution was scaled to have the same integral as the initial distribution. A good agreement between theory and experiment can be seen in figure 5.15 after scaling the DWEIKO prediction to the measured data.

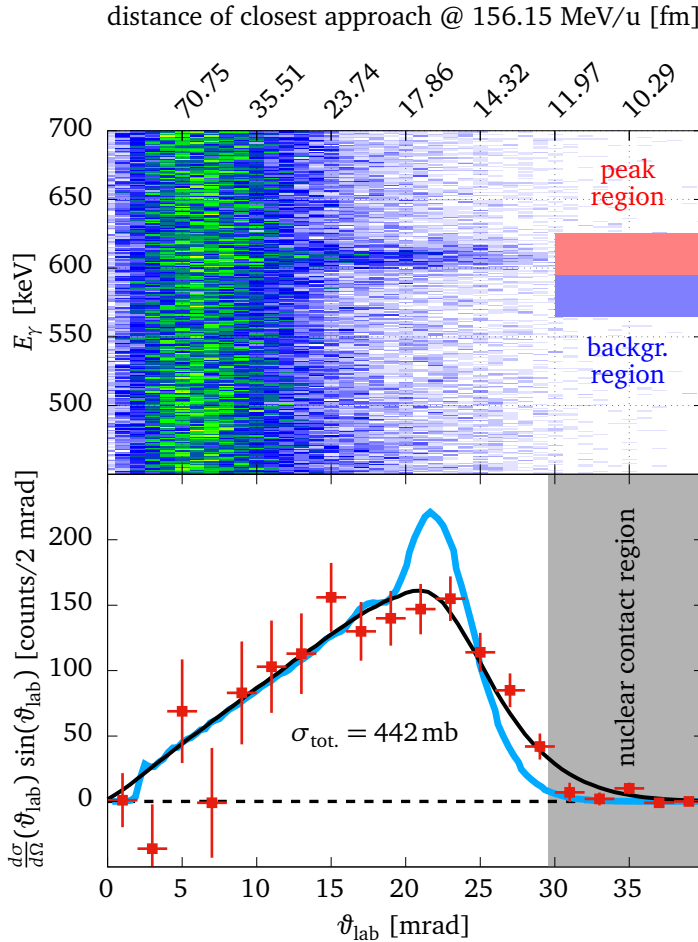


Figure 5.15.: The top histogram shows an E_γ - ϑ_{lab} correlation plot, and the bottom data is the projection of the differential cross section in the energy region of Doppler corrected projectile de-excitation γ -rays. ϑ_{lab} is the particle scattering angle in the laboratory frame as measured by the particle tracking detectors. The experimental values (red boxes) are compared with theoretical calculations with DWEIKO, with (black) and without (blue) taking angular straggling of $\sigma_{\text{strg.}} = 5.5 \text{ mrad}$ into account. The differential DWEIKO cross section was fitted to the projectile data. The width of the optical potential r_0 (see table 5.1) in the DWEIKO calculation was adjusted to fit the data.

5.6.6 Measured total PreSPEC-AGATA efficiency

For the planning of future HISPEC experiments at FAIR, it is important to know the total efficiency $\varepsilon_{\text{total}}$ of the PreSPEC-AGATA setup, in order to make estimates of the required beam time. It is defined by the following expression

$$N_{\gamma} = N_{\text{part.}} \cdot p_{\text{ex.}} \cdot \varepsilon_{\text{total}}, \quad (5.16)$$

with the number N_{γ} of detected γ -rays of interest in the particle-gated, final spectrum, total number $N_{\text{part.}}$ of incident particles that can induce the reaction of interest, the probability $p_{\text{ex.}}$ for the reaction of interest to happen. In other words: the total efficiency is the probability of seeing an event of interest, knowing that the reaction of interest has happened. For the S424 experiment, FRS identification was not needed, therefore the definition only accounts for LYCCA and AGATA, but excludes the FRS identification efficiency. In all cases where the FRS is actually used, the FRS identification probability would have to be accounted for in equation (5.16).

$N_{\text{part.}}$ is measured by counting the signals from the last scintillation detector of the FRS (FRS Sci41 in figure 4.1). These signals are going into a VULOM module with a TRLO firmware [90] of the PreSPEC data acquisition. This module has 64 internal 32-bit software scalers. The scaler values are read out in each event and can be used to get a precise value of $N_{\text{part.}}$. For the Coulomb excitation data set of figure 5.15 this number is $N_{\text{part.}} = 3.56(1) \cdot 10^8$. The value of $p_{\text{ex.}}$ has to be calculated from the Coulomb excitation cross-section and the target thickness of the gold target.

$$p_{\text{ex.}} = \frac{\sigma_{\text{clx.}} D N_A}{A} = 5.13(7) \cdot 10^{-4}, \quad (5.17)$$

with the Coulomb excitation cross-section from the DWEIKO calculation, $\sigma_{\text{clx.}} = 442 \text{ mb}$, the target thickness $D = 380(5) \text{ mg/cm}^2$, Avogadro's number N_A , and the atomic mass number of the target material $A = 197 \text{ g/mol}$. From the Doppler corrected peak area in figure 5.15, $N_{\gamma} = 1501(89)$ is obtained. Thus, the measured total efficiency of the PreSPEC setup is

$$\varepsilon_{\text{total}} = 0.0082(5) \quad \text{or} \quad 0.82(5)\%. \quad (5.18)$$

Note that this number is for the simplified tracking scheme. A small improvement can be expected if a real tracking algorithm is used. A comparison of the efficiency with different types of tracking parameters for source measurements was done by Lalović [75]. The improvement for in-beam measurements should be similar.

Efficiency for low and high rate runs

The data in figure 5.12 is the sum of run 14 and run 20. The main difference between these runs was the primary beam intensity and the γ -trigger thresholds. From the projectile peak-area in the individual runs, values for $\varepsilon_{\text{total}}$ can be extracted for different rates:

$$N_{\text{part.,14}} = 1.10(1) \cdot 10^8, \quad N_{\gamma,14} = 585(48), \quad \varepsilon_{\text{total,14}} = 1.03(8)\% \quad (5.19)$$

$$N_{\text{part.,20}} = 2.46(1) \cdot 10^8, \quad N_{\gamma,20} = 906(61), \quad \varepsilon_{\text{total,20}} = 0.72(5)\% \quad (5.20)$$

	quantity	run 14	run 20
beam	$N_{\text{part.}}$	$1.100 \cdot 10^8$	$2.456 \cdot 10^8$
	duration	640 min	528 min
	average rate	2756 Hz	7733 Hz
	on-spill rate	≈ 4.5 kHz	≈ 14.5 kHz
lmd only	$N_{\text{trig. 10}} \cdot f_{\text{red.}}$	$7.103 \cdot 10^7$	$1.989 \cdot 10^8$
	$N_{\text{trig. 10}}$	$2.775 \cdot 10^5$	$7.768 \cdot 10^5$
	$N_{\text{trig. 9}}$	$1.364 \cdot 10^7$	$6.684 \cdot 10^6$
after merging	$N_{\text{trig. 10}} \cdot f_{\text{red.}}$	$5.458 \cdot 10^7$	$1.486 \cdot 10^8$
	$N_{\text{trig. 10}}$	$2.132 \cdot 10^5$	$5.803 \cdot 10^5$
	$N_{\text{trig. 9}}$	$1.323 \cdot 10^7$	$6.506 \cdot 10^6$
useful events	$N_{\text{trig. 10}} \cdot f_{\text{red.}}$	$3.254 \cdot 10^7$	$7.290 \cdot 10^7$
	$N_{\text{trig. 10}}$	$1.271 \cdot 10^5$	$2.847 \cdot 10^5$
	$N_{\text{trig. 9}}$	$9.820 \cdot 10^6$	$3.812 \cdot 10^6$

Table 5.2.: Trigger 10 was the particle (Sci41) trigger, reduced by $f_{\text{red.}} = 256$. Trigger 9 was the particle- γ trigger (Sci41,LYCCA(wall DSSSD | ToF-target), γ). Events from AGATA and lmd-file were merged, if and only if particle data and γ data was available. Useful events are defined as events that contain all data to make a full analysis, i.e. a full LYCCA-ID and particle multiplicity one. The lower number of trigger 9 events in the high-intensity run is due to different γ trigger thresholds.

5.6.7 Efficiency considerations

Table 5.2 shows a summary of all numbers of the two runs that are of importance for the following discussion.

Dead-time, rate, and trigger probability

If events are completely random at a given rate, the time difference distribution between successive events has an exponential form, and its mean value is the average time between two events and the inverse of the rate. The distribution of *recorded* events has the same form, but with a sharp cut-off at the dead-time. This is illustrated in figure 5.16, where the relation between event rate and the distribution of time differences between successive events is shown. The spill structure of the beam is clearly visible, and is a result of the SIS18 synchrotron being filled from the UNILAC (off spill) and emptied towards the experiment (on spill). Green and red lines represent the rate of accepted physics triggers (trigger 9: γ -Sci41-LYCCA) at two different runs of the experiment that are different in the primary beam intensity and the γ -trigger thresholds. The distribution on the right hand side of figure 5.16 can be fitted with an exponential function and the rate can be extracted as the parameter in the exponent. The distribution has a sharp cut-off at the DAQ dead-time of $\approx 130\mu\text{s}$, as can be seen in the inset, which shows the low end of the distribution.

Each particle that enters the setup, has a finite probability $p_{\text{trig.}}$ to induce a trigger condition. The dependence of the life-time ratio on the rate R and trigger probability $p_{\text{trig.}}$ can be studied with a simple model, illustrated in figure 5.17, where the dead-time is a fixed value. Events are simulated by drawing $N_{\text{part.}}$ uniformly distributed random particle times within a certain duration D . That corresponds to a beam with rate $R = N_{\text{part.}}/D$. Each particle induces a trigger condition with probability $p_{\text{trig.}}$. Each trigger keeps the system busy for t_{dead} , and any trigger inside that busy period does not trigger again. The sum of all busy periods, divided by the duration of the simulation gives the life-time ratio r_{life} . With this model, the relation between

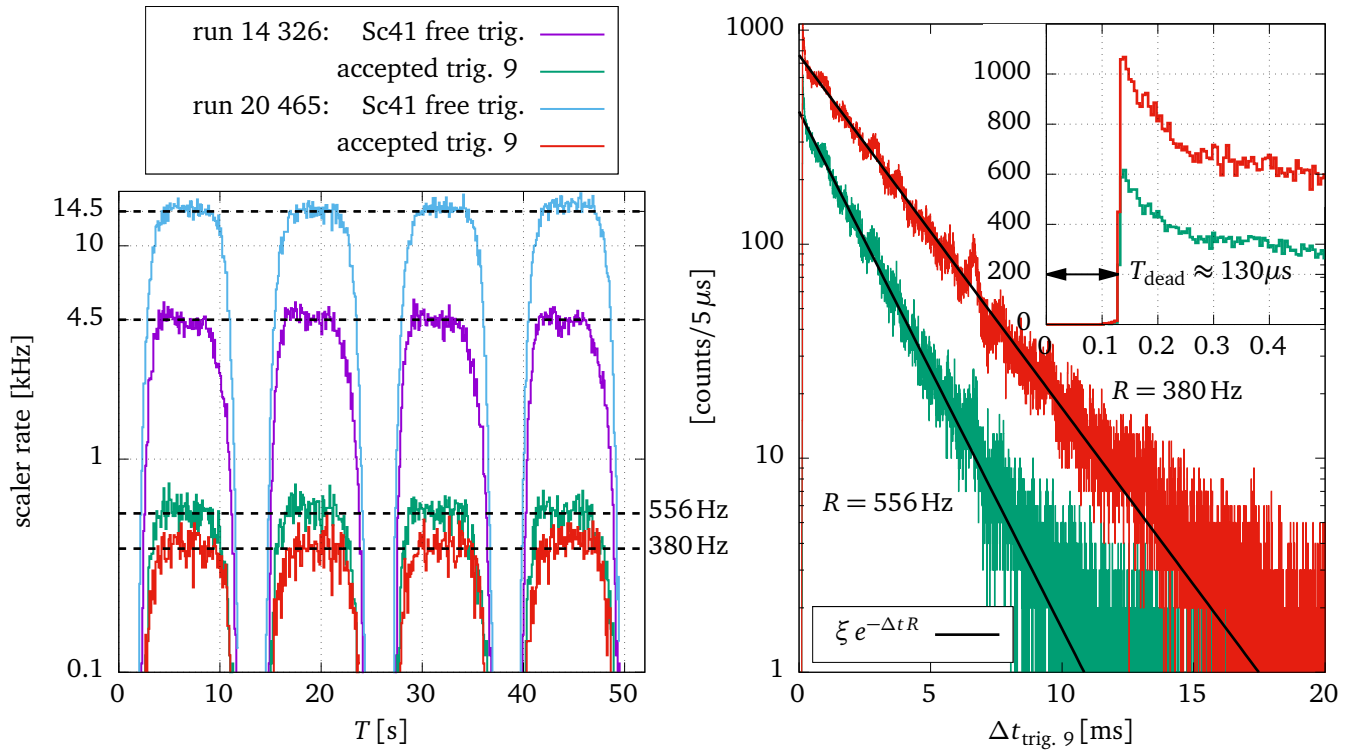


Figure 5.16.: On the left hand side is a plot of the average rate as determined from the scalers inside the TRLO firmware of the VULOM module, shown over a time of 50 seconds. On the right hand side, a histogram of GTS timestamp differences between successive trigger 9 events is shown, which follows an exponential distribution down to the dead-time of $t_{\text{dead}} \approx 130 \mu\text{s}$.

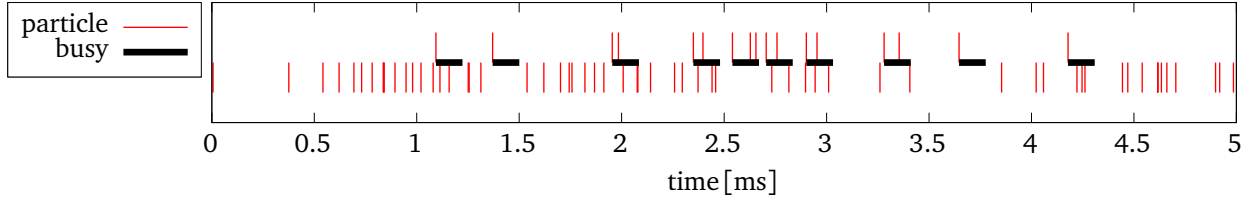


Figure 5.17.: Example of a simulation of triggers and dead-time. The particle rate is $R = 14.5 \text{ kHz}$, the trigger probability per particle is $p_{\text{trig.}} = 0.2$, and the dead-time is fixed to $t_{\text{dead}} = 130 \mu\text{s}$. Each trigger starts a busy period if the system is not already busy. That period has the length of t_{dead} . The events (red vertical lines) in the top row triggered, and the bottom row did not trigger. Top-row events that are inside a busy period, do not induce a new busy period.

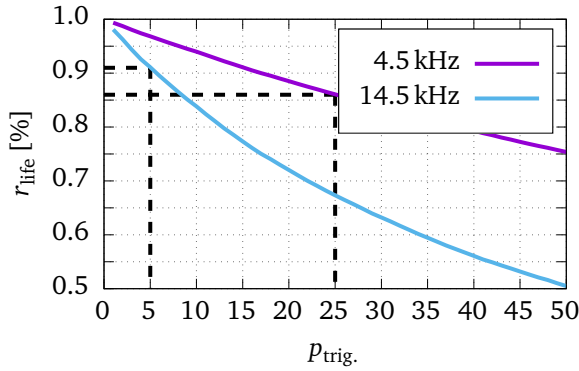


Figure 5.18.: Life-time ratio as a function of the trigger probability for the two particle rates 4.5 kHz (purple) and 14.5 kHz (blue), and a fixed dead-time of $t_{\text{dead}} = 130 \mu\text{s}$. Dashed, black lines indicate the situation for the runs 14 and 20, where the trigger probability must have been different.

R , $p_{\text{trig.}}$, and r_{life} can be studied. Figure 5.18 shows the dependence of the trigger probability on the life-time ratio at two different particle rates. The life-time ratio during the experiment was measured by two scaler channels, one with a 10 kHz pulser signal, the other one with the same signal in anti-coincidence with the DAQ-busy signal. The ratio of these two scaler values is directly the life-time ratio of the DAQ. This is shown in figure 5.19 for the two different runs, and the results are $r_{\text{life},14} \approx 0.86$, and $r_{\text{life},20} \approx 0.91$. With the simple model described above, figure 5.18 can be used to estimate that the trigger probability was $p_{\text{trig.},14} \approx 25\%$ for run 14, and $p_{\text{trig.},20} \approx 5\%$ for run 20. The prediction of trigger rate from the simulation for run 14 with

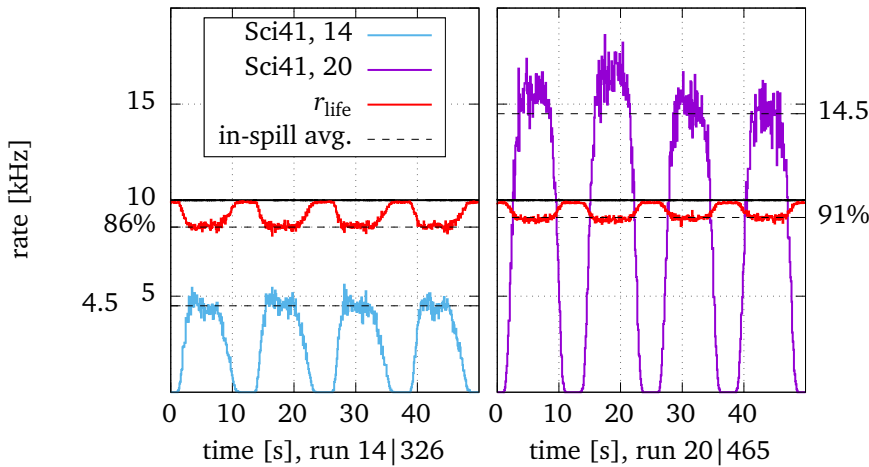


Figure 5.19.: Life-time ratio r_{life} , measured by a 10 kHz pulser in anti-coincidence with the DAQ-busy signal (red).

a particle rate of $R_{14} \approx 4.5 \text{ kHz}$, and a trigger probability of $p_{\text{trig.}} = 0.25$ is $R_{14} \cdot p_{\text{trig.}} = 1125 \text{ Hz}$. For run 20, the numbers are $R_{20} \approx 14.5 \text{ kHz}$, $p_{\text{trig.}} = 0.05$, and $R_{20} \cdot p_{\text{trig.}} = 725 \text{ Hz}$. In the

experiment, for the runs 14 and 20, approximately half of the triggers had number 9 (Sc41- γ -LYCCA), i.e. the results have to be divided by two. The final results are thus $r_{\text{trig},9,14} = 562 \text{ Hz}$, and $r_{\text{trig},9,20} = 362 \text{ Hz}$, in good agreement with the observed trigger 9 rates in figure 5.16. This discussion is based on a simplified model with a fixed dead-time. In reality, the dead-time depends on the number of data words read by the data acquisition system and varies with the event size. However, this simple model confirms the consistency of the written data. It also shows that such a simple model can be used to plan experiments, in particular to estimate the data rate that is produced by an experiment.

Another simplification in the model is the assumption of an exponential distribution of the time between two successive particles, i.e. that the particle arrival time is uncorrelated to previous particles. That this is not the case can be seen by estimating the life-time ratio in a different way. Assuming random particles, the life-time ratio can be defined as the fraction of particle triggers that happen outside a dead-time period and the total number of particles. The former can be obtained by the number of reduced particle triggers N_{red} , times the reduction factor f_{red} , and the latter can be obtained from the hardware scalers of the particle triggers: $r_{\text{part.-life}} = f_{\text{red}} \cdot N_{\text{red}} / N_{\text{part.}}$. The reduced particle trigger is the highest priority trigger (10), is created for one particle in f_{red} , with $f_{\text{red}} = 256$. The reason for the reduction of the trigger 10 frequency is to reduce the data rate to an acceptable level. Results for the estimate of the life-time ratio in this way is $r_{\text{part.-life},14} = 0.646$ for run 14 and $r_{\text{part.-life},20} = 0.801$ for run 20. Both numbers are below the life-time ratio as measured in figure 5.19: $r_{\text{part.-life},14} = 0.75 r_{\text{life},14}$, and $r_{\text{part.-life},20} = 0.88 r_{\text{life},20}$. This indicates, that particles are *not* randomly distributed in time. There is an increased probability to see a particle if another particle was seen before, i.e. more particles than expected arrive within one dead-time period. The result is an effective *decrease* of the efficiency. The deviation of the particle time difference distribution from an exponential form can also be seen in 5.16 which is not exponential towards $\Delta t = 0$. More such histograms can be seen in [66]. Such a behavior is bad for experiments with triggered data acquisition systems and where particles have to be identified one-by-one: first, more particles will arrive inside the busy window of the DAQ, and second, more events will have to be rejected because of multiple particles. The degree of non-randomness was different for different runs, and does apparently not necessarily get bigger for higher beam rates. The mechanism that leads to this particle correlation is not known, and any further discussion is beyond the scope of this work.

Data merging efficiency

Events might get lost while merging data from MBS and AGATA. This can be estimated by determining the number of trigger 9 events (particle- γ trigger) in the MBS data alone, and compare it to the number of trigger 9 events with at least one matching GTS timestamp in the AGATA data. The ratio of these two numbers represents the data merging efficiency. For the relevant data set it was determined as $\varepsilon_{\text{merge}} = 0.9699$, or 97%. Ideally, that number should to be 100%. The reason for the 3% loss was not investigated. Doing so would involve the full AGATA data acquisition system and is beyond the scope of this work.

Data selection efficiency

Events are discarded if there was more than one particle detected, or if for any other reason the particle- γ information is not complete. This is caused by the limited efficiency of all detectors that participate in the particle identification. It can be calculated as the ratio of all trigger 9 events that contribute to the final analysis, and all recorded trigger 9 events. This is called event

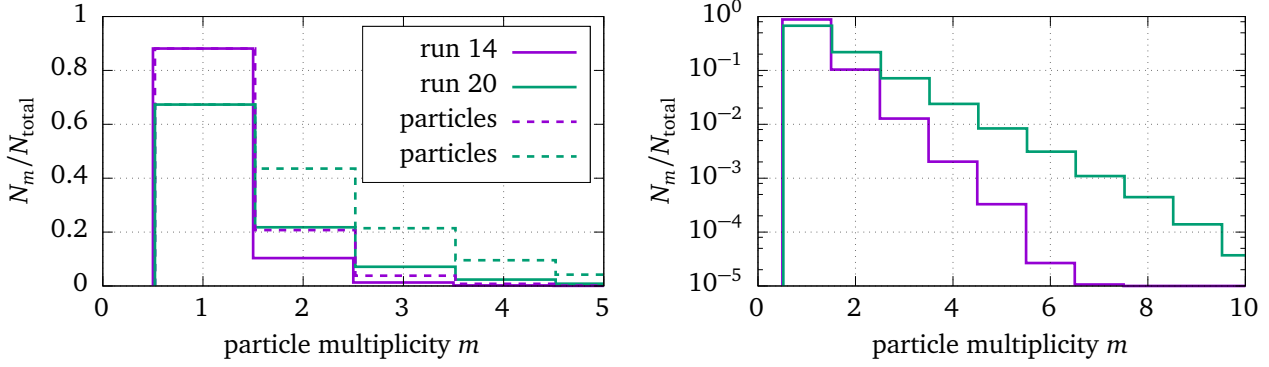


Figure 5.20.: Particle multiplicity as obtained from the LYCCA ToF-stop detector. The histograms are normalized to the number of particles, and follow very precisely an exponential distribution, as can be seen on the logarithmic plot on the right hand side. Dashed lines show the number of particles, i.e. the multiplicity times the frequency of the event.

selection efficiency and was determined to be $\varepsilon_{\text{select}} = 0.718$. This number was significantly different for the two runs, AGATA-14 and AGATA-20: $\varepsilon_{\text{select},14} = 0.743$ and $\varepsilon_{\text{select},20} = 0.585$. The main difference in those two runs was the average primary particle rate: $\langle R_{14} \rangle = 2.76$ kHz and $\langle R_{20} \rangle = 7.73$ kHz. The higher rate in run 20 lead to a lower fraction of events with particle multiplicity one. Figure 5.20 shows the actual distribution of particle multiplicities in runs 14 and 20, obtained from the multiplicity in LYCCA ToF-stop scintillator. From these histograms, the fraction of multiplicity one events can be estimated by

$$r_{m1} = \frac{\sum_{m>1} m \cdot N_m}{\sum_{m>0} m \cdot N_m}, \quad (5.21)$$

where r_{m1} is the relative frequency of the respective multiplicity m . For run 14, $r_{m1,14} = 0.775$, and for run 20, $r_{m1,20} = 0.451$ which is in reasonable agreement with the numbers of $\varepsilon_{\text{select}}$. This shows, that the multiplicity one criterion is the most important one, especially for high intensity runs. A major gain in the total efficiency can be expected if the analysis could recover the particle- γ correlation from the events with multiplicity $m > 1$. The success of experiment S426 depends on the development of such an improvement.

γ -ray detection efficiency

Only a fraction of the emitted γ -radiation is detected inside AGATA, which is determined by the photo-peak efficiency of the detector array ε_γ . In this case, it is the sum of single crystal efficiencies, because the simplified tracking scheme was applied. This single crystal efficiency has to be multiplied with the number of crystals and folded with the angular distribution curve and the Doppler shifted energy distribution. Doing this in a numerical way (see section 5.6.4), results in a value of $\varepsilon_\gamma = 0.0247$ for the projectile γ -radiation.

LYCCA efficiency

Just as the γ -efficiency, the LYCCA efficiency is assumed to be independent of the particle rate. The LYCCA efficiency is mainly determined by the area between the active LYCCA wall modules. Simple geometrical estimate would be the ratio of the active area and the total covered area.

Based on the numbers in [23], with an active area per module of $a = 58.5$ mm, and a gap size of 3.8 mm, the geometric efficiency is $\varepsilon_{\text{LYCCA}} \approx \varepsilon_{\text{LYCCA,geom.}} = a^2/(a + g)^2 = 0.88$. This calculation is simplified in that the beam profile on the LYCCA wall is not taken into account. However, it is a reasonable upper limit for the LYCCA efficiency.

Calculated efficiency

The total PreSPEC-AGATA efficiency $\varepsilon_{\text{total}}$ should be equal to the product of all aforementioned factors, and indeed

$$r_{\text{part.-life,14}} \varepsilon_{\text{m1,14}} \varepsilon_{\gamma} \varepsilon_{\text{LYCCA}} = \varepsilon_{\text{total,14}} \\ 0.646 \cdot 0.775 \cdot 0.0247 \cdot 0.88 = 0.0109 = 1.09\% \quad (\text{measured: } 1.03(8)\%) \quad (5.22)$$

$$r_{\text{part.-life,20}} \varepsilon_{\text{m1,20}} \varepsilon_{\gamma} \varepsilon_{\text{LYCCA}} = \varepsilon_{\text{total,20}} \\ 0.801 \cdot 0.451 \cdot 0.0247 \cdot 0.88 = 0.0079 = 0.79\% \quad (\text{measured: } 0.72(5)\%) \quad (5.23)$$

are in reasonable agreement with the measured values. That shows that all major contributions to the efficiency loss were identified and can be considered in future experiments. Note that neither the calculation, nor the measurement was using any of the FRS detectors. The FRS detector efficiency has to be considered as well for all experiments where incoming particle identification is required. The main contribution of the data analysis is the factor ε_{m1} . It might be possible to find the correct particle- γ correlation within a certain fraction of all multi particle events. This can increase the efficiency of the analysis in particular for experiments with higher rates.

5.6.8 Nuclear excitation from particle detectors

This section discusses the two small peaks right next to the Doppler corrected 2_1^+ de-excitation peak in figure 5.15. In that figure, they are labeled DSSSD nuclear excitation and ToF nuclear excitation, respectively. These are also de-excitation γ -rays from the 2_1^+ state in ^{80}Kr , but the excitation took place not in the gold target but in the detector material before the gold target. This effect was studied in by looking at another run with a 150 mg/cm^2 beryllium secondary target, and a LYCCA gate on the ^{78}Kr fragment, where the 2_1^+ state is at 455 keV. Because of the close distance of the AGATA crystals to the target, and the position resolution of the detected γ -rays, it is possible to determine the origin of gamma rays emitted in-flight. The decay position is a parameter in the Doppler correction formula. In figure 5.21, this parameter was varied and a Doppler corrected spectrum was created for each value of this parameter. Peaks appear at positions that correspond to decay positions of the projectile. The PreSPEC target chamber contained the following pieces of matter: the LYCCA ToF-target detector (50 mg/cm^2 thick plastic scintillator), the LYCCA DSSSD (70 mg/cm^2 thick Si), and the secondary target (150 mg/cm^2 thick Be). This is illustrated in figure 5.21 (top left). On the right hand side of that figure, a z-position scan can be seen, where the assumed projectile decay position was varied along the beam axis while doing the Doppler correction of the γ -radiation. Each line of the 2-dimensional histogram represents the Doppler corrected spectrum using the respective decay position. Peaks are forming at the location of the detector and target material. The γ -yield corresponds to the fragmentation cross-section at the different targets. This phenomenon has applications in other experiments, e.g. the PreSPEC experiment S426 relies on the fact that

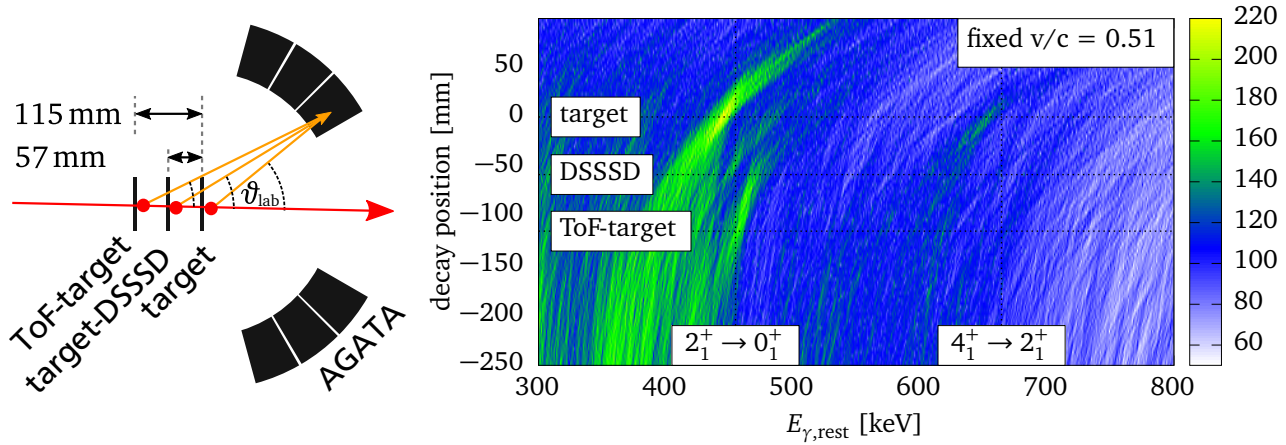


Figure 5.21.: The fragmentation reaction (^{80}Kr , ^{78}Kr 2n) happens not only in the Be target, but also in the detector material. γ -rays of excited states in ^{78}Kr will be emitted shortly after the excitation point, resulting in different emission angles for the three materials. The histogram shows the Doppler corrected γ energy for different assumed decay positions. Peaks form at the positions of the matter in the target chamber. The peaks are at higher energy for earlier excitation positions, because the projectile velocity was higher at the earlier targets.

the distinction of different decay points is possible. In this particular experiment, the Coulomb excitation cross section is measured in two targets simultaneously. For more details refer to [66] and [91].

5.6.9 Discussion

The results of the analysis demonstrate that the measured data is consistent with theoretical predictions. Reasons for efficiency losses were identified and are understood. The analysis shows, that Coulomb excitation experiments can be quantitatively evaluated using the gold target as reference point, rather than different projectiles. This was already done in other experiments such as presented in [88], but the projectile energy was significantly lower ($\approx 80 \text{ MeV/u}$). This is the first time that it was shown that the procedure works also at higher energies around 150 MeV/u using HPGe detectors for γ -ray detection. Recent, similar experiments at RIKEN, such as [92] on ^{104}Sn or previous fast-beam RISING experiments at GSI (e.g. [93] [94]) used exclusively the relative measurement against a projectile reference. The usage of the target excitation as reference has a number of advantages:

- Only one run is needed to complete the experiment. This is an advantage because in the same time more events can be measured and the resulting error is potentially smaller. Of course, this depends also on the ratio of cross-sections of projectile and target excitation.
- No normalization to incoming beam current is needed. As can be seen in figure 5.16, the relation of the particle- γ trigger rate is not always trivial. The relative measurement to the target completely excludes the different beam current as a source of uncertainty.
- Target and projectile excitation happens under the exact same conditions.

The technique of in-beam γ -ray spectroscopy is not new. Neither is the relative measurement of Coulomb excitation cross-sections of fast beams relative to the target excitation. The emphasis of this work has been different: the analysis of the PreSPEC-AGATA commissioning Coulomb excitation data shows for the first time a complete and consistent picture of the relevant processes, including the understanding of the overall efficiency of the experimental setup. The major impact of the efficiency loss due to multi-hit events is pointed out and quantified. Future developments should have a focus on developing techniques to reconstruct these events. It was shown that the measurement of Coulomb excitation cross-sections relative to target excitations is possible for experiments after the GSI Fragment separator with the AGATA spectrometer and beam velocities at $v \approx 0.5c$. In all previous experiments of the same type at GSI, the measurements were done relative to a second beam. Further, it was shown that the main contribution of particle-background comes from fragmentation reactions inside the detector and target material.

The presented analysis could be improved in the following ways:

- A gate on the scattering angle can be applied to achieve better peak to background ratio in the projectile and target de-excitation lines. This might result in a more accurate experimental cross-section ratio. Because of the low number of counts in the target de-excitation peak, further reducing this yield by such a cut might also produce a less accurate result. However, if such a cut is done, the theoretical cross section ratio has to be calculated based on the differential cross-section curves (see figure 5.15) with similar cuts.
- A careful optimization of the AGATA pulse-shape analysis might result in a better energy and time resolution. Tools for the replay are available. Such replays have been done by Michelagnoli [74] Stahl [66] and Ralet [53].
- Another possible improvement is an optimization of the target and detector geometry in the analysis. Stahl has shown in [66] that a numerical search for the best parameters is possible if a clear peak is visible. Such a procedure could also be applied to the data of the S424 experiment. This might result in a considerable improvement of resolution of the Doppler-corrected projectile peak. However, it will change nothing about the uncertainty in the target peak, which is the dominating source of uncertainty in the cross section ratio.
- A large fraction of events have to be discarded because of a particle multiplicity larger than one. In principle, it should be possible to establish γ -particle correlations between the multiple detected particles and recover a certain fraction of these events. One way of achieving this could be the following: First, the position sensitivity of the LYCCA ToF detectors can be exploited to correlate the particle time at the LYCCA wall with the $E-\Delta E$ information. The LYCCA (A,Z) information could then be assigned to both particles. The AGATA time resolution is good enough to assign measured γ -ray times to the respective particle information and separate a multiplicity two event into two multiplicity one events.

A Getting started with PreSPEC data analysis

In this chapter, I will try to summarize useful information for people who want to analyze data from the PreSPEC campaign.

A.1 Raw data access

If you do not have access to the raw data of the experiment, you need to contact someone from the GSI Gamma-Spectroscopy group. At GSI, all data from the PreSPEC experiment is stored on the tape archive. The archive name is `prespecraw`. The archive can be accessed using the `gstore` program. Please read the manual of that program <http://web-docs.gsi.de/~expdv/gstoreManPage.pdf>¹. The location of that program on any GSI Linux machine is² `/cvmfs/it.gsi.de/gstore/gstore64` or `/cvmfs/it.gsi.de/gstore/gstore`³. The `prespecraw` archive is accessible from the GSI user account `agatadaq`. You have to be logged in on that account to do the retrieval from the archive. Typical commands are

- `/cvmfs/it.gsi.de/gstore/gstore64 query "*.lmd" prespecraw "**/*"`
Look for all MBS .lmd files inside the `prespecraw` archive that are located in any subdirectory in any subdirectory of the root directory. This will find for example `/prespecraw/mar_2014/data/85Br_coulex_AG75_1367.lmd` and `/prespecraw/feb_2014/calib/Co60_hector_agata_pos1_0281.lmd` and many more
- `/cvmfs/it.gsi.de/gstore/gstore64 query "*.tar" prespecraw "**/*"`
Look for all .tar files (that contain all AGATA raw data from a run) inside the `prespecraw` archive that are located in any subdirectory in any subdirectory of the root directory. This will find for example `/prespecraw/mar_2014/agata/run_0075_85Br_coulex.tar` and `/prespecraw/feb_2014/agata/run_0011_Fe54_isomer.tar` and many more.
- `/cvmfs/it.gsi.de/gstore/gstore64 retr /lustre/nyx/gamma/46Cr_coulex_AR51_0388.lmd prespecraw apr_2014/data/`
This will retrieve the file `46Cr_coulex_AR51_0388.lmd` from the archive `prespecraw` in the subdirectory `apr_2014/data/`, and copy it to the destination `/lustre/nyx/gamma/46Cr_coulex_AR51_0388.lmd`⁴.

¹ In case that link is broken as you read this, try to search for “gsi gstore manpage” in the world wide web.

² At the time of this writing.

³ In case you are working on a computer that still has no 64 bit operating system.

⁴ The directory `/lustre/nyx/gamma/` refers at the time of this writing to the GSI Gamma-Spectroscopy directory on the mass storage disks of the GSI Green Cube cluster computer. If directory names have changed by the time you are reading this, consult the GSI IT department. However, the tape archive directory names should not change.

A.2 Match MBS and AGATA data files

For each AGATA run you have to get all `.lmd` files and the corresponding `.tar` file with all the AGATA files. Most `.lmd` files have the AGATA run number in the filename (e.g. `AR71`), but this is not guaranteed to be correct. In order to find the correct set of `.lmd` files, the GTS timestamps have to be compared. The prespec software package contains tools to extract the first and last GTS timestamp from an `.lmd` file or an AGATA run directory. These tools are called `gts_timestamp_mbs` and `gts_timestamp_adf`, respectively. For the timestamp in the MBS (`.lmd`) file, just pass the filename as command line argument. For the timestamp of an AGATA run, you have to pass the directory name `path/to/agata_run_XXXX/Data` as command line argument to the program. The `/Data` directory contains one subdirectory for each AGATA crystal in that run. Both programs will output the input name, a time information of the file/directory, and two numbers that are the first and last GTS timestamp in the respective file/directory. You should call that program with all `.lmd` files of the respective experiments and all AGATA runs. With the resulting GTS timestamps, you can find the matching `.lmd` files for each AGATA run.

A.3 Access to the software tools

The tools are installed under the `agata14` account at GSI. You can also obtain the prespec software package by cloning the GIT repository: `git clone agatagsi@lx-pool.gsi.de:git/prespec`. For the password, you have to consult someone from the Pre-SPEC collaboration or someone from the GSI Gamma-Spectroscopy group. A requirement of the `prespec` software package is the `eventapiV50` library. You can find this from the home directory of the `agatagsi` or `agata14` user accounts at GSI. Copy the `eventapiV50` directory to your machine, enter the directory and type `GSI_OS=Linux make`. This should compile the library. Then, the prespec toolset can be built using the GNU autotools⁵. Type `./autogen.sh --prefix=/install/directory --with-gsievlib-path=/path/to/eventapiV50 --with-gsievlib-include-path=/path/to/eventapiV50`, then type `make install`. The tools `preplay`, `gts_timestamp_mbs`, and `gts_timestamp_adf` should be installed now. `preplay` is a command line tool for data analysis. If you want to see histograms as the data is replaying (or want to draw gates on particle-ID) you need to install the Go4 program www.gsi.de/en/work/fairgsi/rare_isotope_beams/electronics/data_processing/data_analysis/the_go4_home_page.htm⁶. In the prespec software package, you find a subdirectory `examples/Go4prespec`. After both, Go4 and the prespec package are successfully installed, enter that directory and type `make`. This should compile the Go4 plugin to do prespec analysis in Go4. The prespec software package contains a subdirectory `doc/PreSPEC_Tutorial` with a tutorial document. Please read that document to get started using the package. It is also helpful to read the first chapter of this work to get familiar with the concepts of the framework, but keep in mind that the Elder framework as described in this work is not identical to the prespec software package. There are slight differences in the interface and in the syntax of the Elder-DSL scripts.

⁵ It might be helpful to make yourself familiar with the GNU autotools

⁶ if that link does not work when you read this, search the world wide web for: GSI Go4 download

A.4 Elder-DSL script for the experiment

Usually during the campaign, snapshots of the analysis scripts have been made towards the end of the experiment. Inside the `agata14` home directory, they are in the `experiments` subdirectory, for example `experiments/43Ti_isomers/offline/analysis.config` is the top level analysis configuration for the ^{43}Ti isomer analysis. The configurations are all very similar, and you can choose as starting point for your analysis any of them. Preferably, you will chose one that was saved during the time when the data for your analysis was taken.

A.5 Online PreSPEC forum

In case of problems, you can look in the `NUSTAR/PRESPEC` forum on the website <https://forum.gsi.de/index.php>. Many questions have already been asked and answered. In case the forum looks inactive, try to write an email to the forum members that were active in the past. Not everybody is checking the forum everyday, especially if it was inactive for some time. However, please ask all your questions in the forum, not only by mail. The more questions are answered publicly in the forum, the greater is the chance that the next person will find an answer to his/her question there.

A.6 Root trees

Most people will prefer to have the final steps of the analysis in their own Root script. It is possible to create a Root tree with a subset of the data on the flow graph. Such a tree can be created using the `preplay` tool. Just add the name of a processor node (`MyProcessor` in this case) via the `--root-add-processor=MyProcessor` command line argument list when starting `preplay`. This will add all inputs and all outputs of `MyProcessor` as leaves to the Root tree. There is no upper limit in the number of processor names that can be added to the Root tree. If you want to have only a single value of `MyProcessor` you can achieve this with following code⁷:

```
1 using UTILS libprespecUTILS.so
2 processor MyValue UTILS.SingleValue
3 value <- MyProcessor.particularValue
4 end
```

Listing A.1: Using an extra processor to put a single value of any processor into a Root tree

and add `--root-add-processor=MyValue` to the argument list of `preplay`. For any array, do the same but use the `UTILS.SingleArray` processor type.

A.7 Developments

In my opinion, most experiments⁸ will require further developments of the analysis algorithms. You can create a local library (say `libpresepMYLIB.so`) containing processors with improved

⁷ Note that this is the syntax for the prespec package, and that there is a difference in the `using` key compared to the Elder-DSL examples in the first chapter.

⁸ in particular for the ones with higher masses or low cross sections

algorithms (say `MyBetterDsssd`). If you come up with a better version of an analysis algorithm, e.g. a better DSSSD analysis, it is probably best to put it in a processor that has a compatible interface to the existing one, i.e. same inputs and outputs as far as possible. Any interface compatible processor can just replace the existing one. Note that it is not possible that two processors have the same name, even if they are in different libraries. In such a case, only one of the two processors is used, with potentially very confusing consequences. This is a shortcoming of the prespec package that was improved in the Elder package [24].

A.8 AGATA PSA replay

It is advisable for a final analysis to redo the AGATA calibration and pulse-shape processing. This was not done in this work and I cannot give too much advice on this topic. The pulse-shape processing might have been improved since the PreSPEC experiments. Make sure, that you have access to the latest versions of the algorithms. Contact the AGATA collaboration for information about the status of these developments.

Bibliography

- [1] E. Rutherford.
The scattering of alpha and beta particles by matter and the structure of the atom.
Philosophical Magazine, volume 21, (1911), pp. 669.
doi:10.1080/14786440508637080.
<http://dx.doi.org/10.1080/14786440508637080>
- [2] J. J. Thomson.
Bakerian Lecture:—Rays of positive electricity.
Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, volume 89(607), (1913), pp. 1. ISSN 0950-1207.
doi:10.1098/rspa.1913.0057.
<http://rspa.royalsocietypublishing.org/content/89/607/1>
- [3] J. Chadwick.
The existence of a neutron.
Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, volume 136(830), (1932), pp. 692. ISSN 0950-1207.
doi:10.1098/rspa.1932.0112.
<http://rspa.royalsocietypublishing.org/content/136/830/692>
- [4] K. Alder, A. Bohr, T. Huus, B. Mottelson, and A. Winther.
Study of Nuclear Structure by Electromagnetic Excitation with Accelerated Ions.
Rev. Mod. Phys., volume 28, (1956), pp. 432.
doi:10.1103/RevModPhys.28.432.
<http://link.aps.org/doi/10.1103/RevModPhys.28.432>
- [5] K. Alder and A. Winther.
Electromagnetic excitation: theory of Coulomb excitation with heavy ions.
North-Holland Pub. Co., 1975. ISBN 9780720402889.
<http://books.google.de/books?id=x-bvAAAAMAAJ>
- [6] G. Gilmore.
Practical gamma-ray spectrometry.
Wiley, 2008. ISBN 9780470861967.
<https://books.google.de/books?id=mNHvAAAAMAAJ>
- [7] C. L. McClelland and C. Goodman.
Excitation of Heavy Nuclei by the Electric Field of Low-Energy Protons.
Phys. Rev., volume 91, (1953), pp. 760.
doi:10.1103/PhysRev.91.760.
<https://link.aps.org/doi/10.1103/PhysRev.91.760>

-
- [8] G. Huber, F. Touchard, S. Büttgenbach, C. Thibault, R. Klapisch, et al.
Spins, magnetic moments, and isotope shifts of $^{21-31}\text{Na}$ by high resolution laser spectroscopy of the atomic D_1 line.
Phys. Rev. C, volume 18, (1978), pp. 2342.
doi:10.1103/PhysRevC.18.2342.
<https://link.aps.org/doi/10.1103/PhysRevC.18.2342>
- [9] D. Guillemaud-Mueller, C. Detraz, M. Langevin, F. Naulin, M. de Saint-Simon, et al.
 β -Decay schemes of very neutron-rich sodium isotopes and their descendants.
Nuclear Physics A, volume 426(1), (1984), pp. 37 . ISSN 0375-9474.
doi:[https://doi.org/10.1016/0375-9474\(84\)90064-2](https://doi.org/10.1016/0375-9474(84)90064-2).
<http://www.sciencedirect.com/science/article/pii/0375947484900642>
- [10] A. Watt, R. P. Singhal, M. H. Storm, and R. R. Whitehead.
A shell-model investigation of the binding energies of some exotic isotopes of sodium and magnesium.
Journal of Physics G: Nuclear Physics, volume 7(7), (1981), p. L145.
<http://stacks.iop.org/0305-4616/7/i=7/a=004>
- [11] M. G. Mayer.
Nuclear Configurations in the Spin-Orbit Coupling Model. I. Empirical Evidence.
Phys. Rev., volume 78, (1950), pp. 16.
doi:10.1103/PhysRev.78.16.
<https://link.aps.org/doi/10.1103/PhysRev.78.16>
- [12] O. Haxel, J. H. D. Jensen, and H. E. Suess.
On the "Magic Numbers" in Nuclear Structure.
Phys. Rev., volume 75, (1949), pp. 1766.
doi:10.1103/PhysRev.75.1766.2.
<https://link.aps.org/doi/10.1103/PhysRev.75.1766.2>
- [13] E. Wigner.
On the Structure of Nuclei Beyond Oxygen.
Phys. Rev., volume 51, (1937), pp. 947.
doi:10.1103/PhysRev.51.947.
<https://link.aps.org/doi/10.1103/PhysRev.51.947>
- [14] T. Otsuka.
Shell, Shape and Spin/Isospin Structures of Exotic Nuclei.
Progress of Theoretical Physics Supplement, volume 146, (2002), pp. 6.
doi:10.1143/PTPS.146.6.
<http://dx.doi.org/10.1143/PTPS.146.6>
- [15] T. J. M. Symons, Y. P. Viyogi, G. D. Westfall, P. Doll, D. E. Greiner, et al.
Observation of New Neutron-Rich Isotopes by Fragmentation of 205-MeV/Nucleon ^{40}Ar Ions.
Phys. Rev. Lett., volume 42, (1979), pp. 40.
doi:10.1103/PhysRevLett.42.40.
<https://link.aps.org/doi/10.1103/PhysRevLett.42.40>

-
- [16] B. Sherrill, D. Morrissey, J. Nolen, and J. Winger.
The A1200 projectile fragment separator.
Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, volume 56-57, (1991), pp. 1106 . ISSN 0168-583X.
doi:[https://doi.org/10.1016/0168-583X\(91\)95108-P](https://doi.org/10.1016/0168-583X(91)95108-P).
<http://www.sciencedirect.com/science/article/pii/0168583X9195108P>
- [17] H. Scheit, T. Glasmacher, B. A. Brown, J. A. Brown, P. D. Cottle, et al.
New Region of Deformation: The Neutron-Rich Sulfur Isotopes.
Phys. Rev. Lett., volume 77, (1996), pp. 3967.
doi:10.1103/PhysRevLett.77.3967.
<https://link.aps.org/doi/10.1103/PhysRevLett.77.3967>
- [18] T. Glasmacher.
Intermediate-energy Coulomb excitation.
Nuclear Physics A, volume 693(12), (2001), pp. 90 . ISSN 0375-9474.
doi:10.1016/S0375-9474(01)01147-2. Radioactive Nuclear Beams.
<http://www.sciencedirect.com/science/article/pii/S0375947401011472>
- [19] T. Motobayashi, Y. Ikeda, K. Ieki, M. Inoue, N. Iwasa, et al.
Large deformation of the very neutron-rich nucleus ^{32}Mg from intermediate-energy Coulomb excitation.
Physics Letters B, volume 346(1), (1995), pp. 9 . ISSN 0370-2693.
doi:[https://doi.org/10.1016/0370-2693\(95\)00012-A](https://doi.org/10.1016/0370-2693(95)00012-A).
<http://www.sciencedirect.com/science/article/pii/037026939500012A>
- [20] S. Wan.
In-beam γ -spectroscopy with Relativistic Radioactive Ion Beams. Ph.D. thesis, Ruprecht-Karls-Universität Heidelberg, 1999.
<https://web-docs.gsi.de/~gsgweb/publ/docu/thesis/wan.ps-gz>
- [21] H. Wollersheim, D. Appelbe, A. Banu, R. Bassini, T. Beck, et al.
Rare ISotopes INvestigation at GSI (RISING) using gamma-ray spectroscopy at relativistic energies.
Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 537(3), (2005), pp. 637 . ISSN 0168-9002.
doi:10.1016/j.nima.2004.08.072.
<http://www.sciencedirect.com/science/article/pii/S0168900204019588>
- [22] S. Akkoyun, A. Algora, B. Alikhani, F. Ameil, G. de Angelis, et al.
AGATA – Advanced GAMMA Tracking Array.
Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 668, (2012), pp. 26 . ISSN 0168-9002.
doi:10.1016/j.nima.2011.11.081.
<http://www.sciencedirect.com/science/article/pii/S0168900211021516>
- [23] P. Golubev, A. Wendt, L. Scruton, J. Taprogge, D. Rudolph, et al.
The LundYorkCologne Calorimeter (LYCCA): Concept, design and prototype developments for a FAIR-NUSTAR detector system to discriminate relativistic heavy-ion

reaction products.

Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 723, (2013), pp. 55 . ISSN 0168-9002. doi:10.1016/j.nima.2013.04.058.

<http://www.sciencedirect.com/science/article/pii/S0168900213004798>

[24] M. Reese.

Data flow framework for data analysis applications, 2017.

<https://sourceforge.net/projects/elderpt/>

[25] J. Grębosz.

The Cracow code – an interactive method of sophisticated online analysis.

Computer Physics Communications, volume 176(4), (2007), pp. 251 . ISSN 0010-4655. doi:10.1016/j.cpc.2006.09.006.

<http://www.sciencedirect.com/science/article/pii/S0010465506003675>

[26] H. T. Johansson.

UCESB - unpack & check every single bit.

<http://fy.chalmers.se/~f96hajo/ucesb/>

[27] X. Grave, R. Canedo, J. F. Clavelin, S. Du, and E. Legay.

NARVAL a modular distributed data acquisition system with Ada 95 and RTAI. 2005. pp. 119–123.

doi:10.1109/RTC.2005.1547454.

<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1547454>

[28] R. Barth, Y. Du, H. Essel, R. Fritzsche, H. . Göringer, et al.

GSI Multi-Branch System User Manual.

GSI, Gesellschaft für Schwerionenforschung mbH, Planckstraße 1, D-64291 Darmstadt Germany, version: 2.2 edition, 2000.

www.gsi.de/fileadmin/EE/MBS/Gm_mbs_i_2.pdf

[29] M. Cappellazzo.

private communication, 2016.

[30] R. Matzdorf, G. Soff, and G. Mehler.

Elastic collisions of heavy ions at intermediate energies.

Zeitschrift für Physik D Atoms, Molecules and Clusters, volume 6(1), (1987), pp. 5. ISSN 0178-7683.

doi:10.1007/BF01436989.

<http://dx.doi.org/10.1007/BF01436989>

[31] A. Winther and K. Alder.

Relativistic coulomb excitation .

Nuclear Physics A, volume 319(3), (1979), pp. 518 . ISSN 0375-9474.

doi:10.1016/0375-9474(79)90528-1.

<http://www.sciencedirect.com/science/article/pii/0375947479905281>

[32] A. Mariano.

GNU units.

<https://www.gnu.org/software/units/>

-
- [33] T. Fließbach.
Elektrodynamik: Lehrbuch zur Theoretischen Physik II.
Spektrum Akademischer Verlag, 2008. ISBN 9783827420213.
<http://books.google.de/books?id=o4syKAAACAAJ>
- [34] B. Gough.
GNU Scientific Library Reference Manual - Third Edition.
Network Theory Ltd., 3rd edition, 2009. ISBN 0954612078, 9780954612078.
https://www.gnu.org/software/gsl/manual/html_node/
- [35] E. Fehlberg.
Klassische Runge-Kutta-Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme.
Computing, volume 6(1-2), (1970), pp. 61. ISSN 0010-485X.
doi:10.1007/BF02241732.
<http://dx.doi.org/10.1007/BF02241732>
- [36] C. A. Bertulani, A. E. Stuchbery, T. J. Mertzimekis, and A. D. Davies.
Intermediate energy Coulomb excitation as a probe of nuclear structure at radioactive beam facilities.
Phys. Rev. C, volume 68, (2003), p. 044609.
doi:10.1103/PhysRevC.68.044609.
<http://link.aps.org/doi/10.1103/PhysRevC.68.044609>
- [37] A. Aleixo and C. Bertulani.
Coulomb excitation in intermediate-energy collisions .
Nuclear Physics A, volume 505(2), (1989), pp. 448 . ISSN 0375-9474.
doi:http://dx.doi.org/10.1016/0375-9474(89)90385-0.
<http://www.sciencedirect.com/science/article/pii/0375947489903850>
- [38] T. Fließbach.
Quantenmechanik: Lehrbuch zur Theoretischen Physik III.
Spektrum Akademischer Verlag, 2005. ISBN 9783827415899.
https://books.google.de/books?id=EM3_uAAACAAJ
- [39] V. I. Lebedev and D. N. Laikov.
A quadrature formula for the sphere of the 131st algebraic order of accuracy.
Doklady Mathematics, volume 59(3), (1999), pp. 477.
- [40] D. Laikov and C. van Wuelen.
Lebedev-Laikov-Grids.
<http://www.ccl.net/ccl/software/SOURCES/FORTRAN/Lebedev-Laikov-Grids/>
- [41] C. Bertulani.
A computer program for relativistic multiple Coulomb and nuclear excitation .
Computer Physics Communications, volume 116(23), (1999), pp. 345 . ISSN 0010-4655.
doi:http://dx.doi.org/10.1016/S0010-4655(98)00141-6.
<http://www.sciencedirect.com/science/article/pii/S0010465598001416>

-
- [42] C. Bertulani.
DWEIKO.
http://cpc.cs.qub.ac.uk/summaries/ADRN_v1_0.html
- [43] M. Reese.
nuco: Nuclear Coulomb-excitation, 2017.
<https://github.com/miree/nuco>
- [44] N. Pietralla, M. Reese, M. Cortes, F. Ameil, D. Bazzacco, et al.
On the Road to FAIR: 1st Operation of AGATA in PreSPEC at GSI.
EPJ Web of Conferences, volume 66, (2014), p. 4.
doi:10.1051/epjconf/20146602083.
http://www.epj-conferences.org/articles/epjconf/abs/2014/03/epjconf_inpc2013_02083/epjconf_inpc2013_02083.html
- [45] Z. Podolyák, W. Korten, J. Jolie, J. Gerl, B. Rubio, et al.
Technical Proposal for the Design, Construction, Commissioning and Operation of the HISPEC/DESPEC experiment at the Low-Energy Branch of the Super-FRS facility.
Technical report, 2005.
http://personal.ph.surrey.ac.uk/~phs1zp/Tech_Proposal_HISPEC_DESPEC-nocost.pdf
- [46] **FAIR - An International Facility for Antiproton and Ion Research.**
<http://www.fair-center.eu/>
- [47] H. Geissel, P. Armbruster, K. Behr, A. Brünle, K. Burkard, et al.
The GSI projectile fragment separator (FRS): a versatile magnetic system for relativistic heavy ions.
Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, volume 70(14), (1992), pp. 286 . ISSN 0168-583X.
doi:10.1016/0168-583X(92)95944-M.
<http://www.sciencedirect.com/science/article/pii/0168583X9295944M>
- [48] H. Geissel, M. Winkler, H. Weick, K.-H. Behr, G. Münzenberg, et al.
Technical Design Report on the Super-FRS. Technical report, GSI, 2009.
- [49] B. Franczak.
SIS Parameter List. 1987.
www.gsi.de/fileadmin/PSchwab/Dokumentationen_Berichte/Beschleunigeranlage/SIS18/sis18_parameterliste.pdf
- [50] GSI.
Autocad-FRS-CAVE-FRS.
https://www-win.gsi.de/frs-setup/Downloads/\T-Autocad-FRS-CAVE-FRS_04_17_Model.pdf
- [51] R. Lozeva, J. Gerl, M. Górska, I. Kojouharov, S. Mandal, et al.
A novel Calorimeter Telescope for identification of relativistic heavy-ion reaction channels.

Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 562(1), (2006), pp. 298 . ISSN 0168-9002.

doi:<http://dx.doi.org/10.1016/j.nima.2006.02.163>.

<http://www.sciencedirect.com/science/article/pii/S0168900206004669>

- [52] A. Wendt, J. Taprogge, P. Reiter, P. Golubev, H. Grawe, et al.
Isospin symmetry in the sd shell: Transition strengths in the neutron-deficient sd shell nucleus ^{33}Ar .
Phys. Rev. C, volume 90, (2014), p. 054301.
doi:10.1103/PhysRevC.90.054301.
<http://link.aps.org/doi/10.1103/PhysRevC.90.054301>
- [53] D. Ralet.
Lifetime measurement in the mass region $A \sim 100$ with the PreSPEC-AGATA setup.
Ph.D. thesis, Technische Universität, Darmstadt, 2015.
<http://tuprints.ulb.tu-darmstadt.de/4613/>
- [54] G. Guastalla, D. D. DiJulio, M. Górska, J. Cederkäll, P. Boutachkov, et al.
Coulomb Excitation of ^{104}Sn and the Strength of the ^{100}Sn Shell Closure.
Phys. Rev. Lett., volume 110, (2013), p. 172501.
doi:10.1103/PhysRevLett.110.172501.
<http://link.aps.org/doi/10.1103/PhysRevLett.110.172501>
- [55] R. Hoischen, S. Pietri, D. Rudolph, W. Prokopowicz, H. Schaffner, et al.
Fast timing with plastic scintillators for in-beam heavy-ion spectroscopy.
Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 654(1), (2011), pp. 354 . ISSN 0168-9002.
doi:10.1016/j.nima.2011.07.013.
<http://www.sciencedirect.com/science/article/pii/S016890021101415X>
- [56] D. Rudolph, C. Fahlander, P. Golubev, R. Hoischen, V. Avdeichikov, et al.
LYCCA Technical Design Report. Technical report, FAIR-NUSTAR, June 2008.
<http://www.nuclear.lu.se/fileadmin/nuclear/Grundlaeggande/lycca-TDR.pdf>
- [57] K. Olive and P. D. Group.
Review of Particle Physics.
Chinese Physics C, volume 38(9), (2014), p. 090001.
<http://stacks.iop.org/1674-1137/38/i=9/a=090001>
- [58] H. Weick.
ATIMA, calculate ATomic Interaction with MATter (www interface).
<https://web-docs.gsi.de/~weick/atima/>
- [59] N. Colonna, G. Wozniak, A. Veeck, W. Skulski, G. Goth, et al.
Calibration of the response function of CsI(Tl) scintillators to intermediate-energy heavy ions.

Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 321(3), (1992), pp. 529 . ISSN 0168-9002.

doi:10.1016/0168-9002(92)90063-A.

<http://www.sciencedirect.com/science/article/pii/S016890029290063A>

- [60] P. Mastinu, P. Milazzo, M. Bruno, M. D'Agostino, and L. Manduci.
Calibration of CsI(Tl) scintillators for heavy ions ($3 \leq Z \leq 54$) in a wide energy range ($\text{Eu} \leq 60 \text{ MeV/u}$).

Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 338(23), (1994), pp. 419 . ISSN 0168-9002.

doi:10.1016/0168-9002(94)91325-0.

<http://www.sciencedirect.com/science/article/pii/S0168900294913250>

- [61] Saint-Gobain.

Premium Plastic Scintillators.

<http://www.crystals.saint-gobain.com/\uploadedFiles/SG-Crystals/Documents/\SGC%20BC418-420-422%20Data%20Sheet.pdf>

- [62] C. T. I. Manual.

Mod. V1290-VX1290 A/N, 32/16 Ch. Multihit TDC, 2010.

- [63] M. Reese, N. Pietralla, P. Golubev, and D. Rudolph.

Position Sensitivity of LYCCA Time-of-Flight Detectors, volume 2013-1 of *GSI Report*, chapter 1, p. 185 p. GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, 2013. 05p12r8n8.

<https://repository.gsi.de/record/52088>

- [64] A. Wendt.

Isospin symmetry in the sd shell: Coulomb excitation of ^{33}Ar at relativistic energies and the new Lund-York-Cologne-Calorimeter. Ph.D. thesis, Universität zu Köln, 2013.

http://www.ikp.uni-koeln.de/groups/reiter/theses/Wendt_diss.pdf

- [65] M. Reese, J. Gerl, P. Golubev, and N. Pietralla.

Automatic intrinsic calibration of double-sided silicon strip detectors.

Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, (0), (2015), pp. . ISSN 0168-9002.

doi:10.1016/j.nima.2015.01.032.

<http://www.sciencedirect.com/science/article/pii/S0168900215000637>

- [66] C. Stahl.

New methods for the γ -ray spectroscopy with position-sensitive detector systems. Ph.D. thesis, Technische Universität, Darmstadt, 2015.

<http://tuprints.ulb.tu-darmstadt.de/4955/>

- [67] D. Sivia and J. Skilling.

Data Analysis: A Bayesian Tutorial.

Oxford science publications. OUP Oxford, 2006. ISBN 9780198568322.

<https://books.google.de/books?id=zN-yliq6eZ4C>

-
- [68] T. Bayes.
An Essay towards Solving a Problem in the Doctrine of Chances.
Philosophical Transactions of the Royal Society, volume 35, (1763), pp. 370.
doi:10.1098/rstl.1763.0053.
<http://rstl.royalsocietypublishing.org/content/53/370>
- [69] P. S. Laplace.
Memoir on the Probability of the Causes of Events.
Statistical Science, volume 1(3), (1986), pp. pp. 364. ISSN 08834237.
doi:10.1214/ss/1177013621.
<http://www.jstor.org/stable/2245476>
- [70] A. Hoecker, P. Speckmayer, J. Stelzer, J. Therhaag, E. von Toerne, et al.
TMVA - Toolkit for Multivariate Data Analysis.
ArXiv Physics e-prints.
<https://arxiv.org/abs/physics/0703039>
- [71] J. Simpson, J. Nyberg, and W. Korten.
AGATA – Advanced Gamma Tracking Array. Technical report, AGATA Collaboration Council, 2008.
http://npg.dl.ac.uk/AGATA/TDR/TDR_EUJRA.pdf
- [72] **AGATA Collaboration Council.**
http://npg.dl.ac.uk/agata_acc/index.html
- [73] D. Bazzacco.
The Advanced Gamma Ray Tracking Array AGATA.
Nuclear Physics A, volume 746, (2004), pp. 248 . ISSN 0375-9474.
doi:10.1016/j.nuclphysa.2004.09.148.
<http://www.sciencedirect.com/science/article/pii/S0375947404009625>
- [74] C. Michelagnoli.
The lifetime of the 6.79 MeV state in ^{15}O as a challenge for nuclear astrophysics and gamma-ray spectroscopy: a new DSAM measurement with the AGATA Demonstrator array. Ph.D. thesis, Padova University, 2013.
<http://paduaresearch.cab.unipd.it/5898/1/MichelagnoliCaterina.pdf>
- [75] N. Lalović, C. Louchart, C. Michelagnoli, R. Perez-Vidal, D. Ralet, et al.
Performance of the AGATA γ -ray spectrometer in the PreSPEC set-up at GSI.
Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 806, (2016), pp. 258 . ISSN 0168-9002.
doi:10.1016/j.nima.2015.10.032.
<http://www.sciencedirect.com/science/article/pii/S0168900215012395>
- [76] D. Ralet, S. Pietri, Y. Aubert, M. Bellato, D. Bortolato, et al.
Data-flow coupling and data-acquisition triggers for the PreSPEC-AGATA campaign at GSI.
Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 786, (2015), pp. 32 . ISSN 0168-9002.

- doi:10.1016/j.nima.2015.03.025.
<http://www.sciencedirect.com/science/article/pii/S0168900215003332>
- [77] H. Essel and N. Kurz.
The general purpose data acquisition system MBS. 1999. pp. 475–478.
 doi:10.1109/RTCON.1999.842672.
<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=842672>
- [78] H. G. Essel, J. Hoffmann, N. Kurz, R. S. Mayer, W. Ott, et al.
The new data acquisition system at GSI.
 IEEE Trans. Nucl. Sci., volume 43, (1996), pp. 132.
 doi:10.1109/23.486018.
<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=486018>
- [79] J. Hoffmann, N. Kurz, and M. Richter.
TRIVA, VME Trigger Module. GSI, 2003.
www.gsi.de/fileadmin/EE/MBS/GSItrigva3_6.pdf
- [80] H. T. Johansson, A. Henriques, C. Langer, H. Simon, and the R3B-collaboration.
GSI Scientific Report 2010 – TRLO II flexible FPGA trigger control, volume 2011-1 of *GSI Report*. GSI, Darmstadt, 2011.
<http://repository.gsi.de/record/53521>
- [81] O. Stezowski.
GammaWare User's Guide, 3 edition, 2009.
<http://agata.in2p3.fr/doc/GwUserGuide.pdf>
- [82] T. Glasmacher.
Coulomb Excitation at Intermediate Energies.
 Annual Review of Nuclear and Particle Science, volume 48(1), (1998), pp. 1.
 doi:10.1146/annurev.nucl.48.1.1.
<http://dx.doi.org/10.1146/annurev.nucl.48.1.1>
- [83] G. Peilert, H. Stoecker, and W. Greiner.
Physics of high-energy heavy-ion collisions.
 Reports on Progress in Physics, volume 57(6), (1994), p. 533.
<http://stacks.iop.org/0034-4885/57/i=6/a=001>
- [84] J. Bondorf, A. Botvina, A. Iljinov, I. Mishustin, and K. Sneppen.
Statistical multifragmentation of nuclei.
 Physics Reports, volume 257(3), (1995), pp. 133 . ISSN 0370-1573.
 doi:http://dx.doi.org/10.1016/0370-1573(94)00097-M.
<http://www.sciencedirect.com/science/article/pii/037015739400097M>
- [85] H. Geissel, H. Weick, C. Scheidenberger, R. Bimbot, and D. Gardès.
Experimental studies of heavy-ion slowing down in matter.
 Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, volume 195(12), (2002), pp. 3 . ISSN 0168-583X.
 doi:http://dx.doi.org/10.1016/S0168-583X(02)01311-3.
<http://www.sciencedirect.com/science/article/pii/S0168583X02013113>

-
- [86] X. Huang and C. Zhou.
Nuclear Data Sheets for A = 197.
Nuclear Data Sheets, volume 104(2), (2005), pp. 283 . ISSN 0090-3752.
doi:<http://dx.doi.org/10.1016/j.nds.2005.01.001>.
<http://www.sciencedirect.com/science/article/pii/S0090375205000025>
- [87] B. Singh.
Nuclear Data Sheets for A = 80.
Nuclear Data Sheets, volume 105(2), (2005), pp. 223 . ISSN 0090-3752.
doi:<http://dx.doi.org/10.1016/j.nds.2005.06.002>.
<http://www.sciencedirect.com/science/article/pii/S009037520500044X>
- [88] H. Scheit.
Low-Lying Collective Excitations in Neutron-Rich Even-Even Sulfur and Argon Isotopes Studied via Intermediate-Energy Coulomb Excitation and Proton Scattering.
Ph.D. thesis, Michigan State University, 1998.
https://publications.nsl.msueu/thesis/Scheit1998_146.pdf
- [89] K. Siegbahn.
Alpha-, beta- and gamma-ray spectroscopy. Volume 2. North-Holland Pub. Co., 1965.
- [90] H. T. Johansson, M. Heil, B. Löher, H. Simon, and H. Törnqvist.
TRLO II - friendly FPGA trigger control, volume 2014-1 of *GSI Report*, p. 354 p. GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, 2014.
doi:10.15120/GR-2014-1-FG-S-FRS-15.
<http://repository.gsi.de/record/68045>
- [91] C. Stahl, N. Pietralla, G. Rainovski, and M. Reese.
Coulex-multipolarimetry with relativistic heavy-ion beams.
Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, volume 770, (2015), pp. 123 . ISSN 0168-9002.
doi:<http://dx.doi.org/10.1016/j.nima.2014.10.024>.
<http://www.sciencedirect.com/science/article/pii/S0168900214011711>
- [92] P. Doornenbal, S. Takeuchi, N. Aoi, M. Matsushita, A. Obertelli, et al.
Intermediate-energy Coulomb excitation of ^{104}Sn : Moderate E2 strength decrease approaching ^{100}Sn .
Phys. Rev. C, volume 90, (2014), p. 061302.
doi:10.1103/PhysRevC.90.061302.
<http://link.aps.org/doi/10.1103/PhysRevC.90.061302>
- [93] L. A. Banu.
 ^{108}Sn studied with intermediate-energy Coulomb excitation. Ph.D. thesis, Fachbereich Physik der Johannes Gutenberg-Universität in Mainz, 2005.
<https://publications.ub.uni-mainz.de/theses/volltexte/2005/817/pdf/817.pdf>
- [94] T. Saito, N. Saito, K. Starosta, J. Beller, N. Pietralla, et al.
Yrast and non-yrast 2+ states of ^{134}Ce and ^{136}Nd populated in relativistic Coulomb

excitation.

Physics Letters B, volume 669(1), (2008), pp. 19 . ISSN 0370-2693.

doi:<http://dx.doi.org/10.1016/j.physletb.2008.09.027>.

<http://www.sciencedirect.com/science/article/pii/S037026930801160X>